



TUGAS AKHIR - TJ 141502

***Implementasi Metode Object Following pada Robot Service  
untuk Penggunaan di dalam Ruangan***

Muhammad Ardi Pradana  
NRP 07211340000008

Dosen Pembimbing  
Muhtadin, ST., MT.  
Dr. I Ketut Eddy Purnama, ST., MT.

DEPARTEMEN TEKNIK KOMPUTER  
Fakultas Teknologi Elektro  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018



FINAL PROJECT - TJ 141502

***Implementation of Object Following Method on Robot Service  
for Indoor Use***

Muhammad Ardi Pradana  
NRP 07211340000008

Advisors  
Muhtadin, ST., MT.  
Dr. I Ketut Eddy Purnama, ST., MT.

Department of Computer Engineering  
Faculty of Electrical Technology  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018

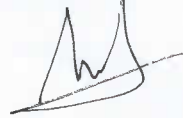
## PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi (sebagian maupun keseluruhan) tugas akhir saya dengan judul "**Implementasi Metode *Object Following* pada *Robot Service* untuk Penggunaan didalam Ruangan**" adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, January 2018



Muhammad Ardi Pradana

NRP. 07211340000008

# LEMBAR PENGESAHAN

## Implementasi Metode *Following Object* pada *Robot Service* untuk Penggunaan didalam Ruangan

Tugas Akhir ini disusun untuk memenuhi salah satu syarat memperoleh  
gelar Sarjana Teknik di Institut Teknologi Sepuluh Nopember Surabaya

Oleh: Muhammad Ardi Pradana (NRP: 07211340000008)

Tanggal Ujian : 04 Januari 2018

Periode Wisuda : Maret 2018

Disetujui oleh:

Muhtadin, ST., MT.

NIP: 198106092009121003

(Pembimbing I)

Dr. I Ketut Eddy Purnama, ST., MT.

NIP. 196907301995121001

(Pembimbing II)

Prof. Dr. Ir. Mauridhi Hery Purnomo.

M.Eng.

NIP: 195809161986011001

(Penguji I)

Arief Kurniawan, ST., MT.

NIP: 197409072002121001

(Penguji II)

Ahmad Zaini, ST., MT.

NIP: 197504192002121003

(Penguji III)

Mengetahui  
Kepala Departemen Teknik Komputer

Dr. I Ketut Eddy Purnama, ST., MT.

NIP. 196907301995121001



# ABSTRAK

Nama Mahasiswa : Muhammad Ardi Pradana  
Judul Tugas Akhir : Implementasi Metode *Object Following*  
pada *Robot Service* untuk Penggunaan di-  
dalam Ruangan  
Pembimbing : 1. Muhtadin, ST., MT.  
2. Dr. I Ketut Eddy Purnama, ST., MT.

Seiring dengan kemajuan teknologi dan ilmu pengetahuan menyebabkan bertambahnya angka usia harapan hidup (UHH) pada masyarakat. Namun bertambahnya angka UHH menyebabkan muncul masalah pada bidang kesehatan. Peningkatan angka UHH tidak diimbangi meningkatnya daya tahan terhadap penyakit. Hal ini dikarenakan kondisi tubuh pada usia lanjut telah mengalami degeneratif (penuaan). Kondisi yang rentan terhadap penyakit ini berakibat pada perlunya pemantauan yang lebih terhadap kaum lanjut usia. Pemantauan terhadap lansia juga harus dilakukan secara intensif untuk mencegah terjadinya hal-hal yang tidak diinginkan. *Robot service* yang tidak memerlukan waktu untuk istirahat dapat menjadi sebuah alternatif solusi untuk melakukan pemantau. *Robot service* dilengkapi dengan sensor yang dapat memonitor kondisi lansia melalui kemampuannya untuk mendeteksi keberadaan seseorang dan mengikuti pergerakan orang tersebut. Agar dapat memantau dan mengikuti orang robot harus mempunyai kemampuan mendeteksi orang. Pada penelitian ini berfokus pada pengimplementasian metode deteksi orang pada robot sehingga robot dapat mengikuti orang tersebut. Metode yang digunakan dalam sistem ini adalah *skeletal tracking*. *skeletal tracking* merupakan metode yang dapat mendeteksi manusia menggunakan sensor depth. Proses deteksi pada *skeletal tracking* meliputi pemisahan latar, kalibrasi, dan tracking. Hasil pengujian dari penelitian ini menunjukkan bahwa robot dapat mendeteksi dan mengikuti seseorang dalam suatu ruangan.

Kata Kunci : Lanjut usia, Robot Pelayan, Deteksi Manusia, *skeletal tracking*.

*Halaman ini sengaja dikosongkan*

# ***ABSTRACT***

*Name* : Muhammad Ardi Pradana  
*Title* : *Implementation of Object Following Method  
on Robot Service for Indoor Use*  
*Advisors* : 1. Muhtadin, ST., MT.  
2. Dr. I Ketut Eddy Purnama, ST., MT.

*As the advances in technology and science lead to an increase in life expectancy (UHH) in the community. But the increase in UHH numbers causes problems in the health field. Increased UHH numbers are not offset by increased resistance to disease. This is because the condition of the body in old age has degenerative (aging). Conditions that are susceptible to this disease result in the need for more monitoring of the elderly. Monitoring of the elderly should also be done intensively to prevent the occurrence of things that are not desirable. A service robot that does not take time to rest can be an alternative solution. Robot service equipped with sensors that can monitor the condition of the elderly through its ability to detect the presence of a person and follow the movement of the person. In order to monitor and follow people robots must have the ability to detect people. In this study focuses on implementing the method of detection of people on robots so that robots can follow the person. The method used in this system is skeletal tracking. skeletal tracking is a method that can detect humans using sensor depth. The detection process on skeletal tracking includes background separation, calibration, and tracking. Test results from this study showed that the robot can detect and follow a person in a room.*

*Keywords :elderly, Robot Waitress, Human Detection, skeletal tracking.*

*Halaman ini sengaja dikosongkan*

# KATA PENGANTAR

Puji syukur penulis ucapkan kehadiran Allah SWT atas segala limpahan karunia-Nya sehingga penulis dapat menyelesaikan tugas akhir dengan judul Implementasi Metode *Following Object* pada *Robot Service* sesuai dengan batas waktu yang telah ditentukan guna memenuhi salah satu syarat kelulusan Program Sarjana Teknik Komputer, ITS.

Dalam pengerjaannya, tugas akhir ini dapat terselesaikan karena bantuan dari berbagai pihak. Oleh sebab itu, dalam kesempatan kali ini penulis mengucapkan terima kasih kepada:

1. Keluarga, Ibu dan Ayah, terima kasih atas dukungan, semangat dan doa yang tidak pernah terhenti.
2. Dr. I Ketut Eddy Purnama, ST., MT. selaku Kepala Departemen Teknik Komputer, Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember.
3. Bapak Muhtadin, ST., MT. dan Dr. I Ketut Eddy Purnama, ST., MT. atas arahan dan bimbingan dalam pengerjaan tugas akhir ini.
4. Dosen-dosen Departemen Teknik Komputer atas ilmu dan bimbingan yang telah diberikan selama kuliah.
5. Enny Amsaro atas dukungan, bantuan, motivasi, saran serta doa yang tidak pernah lupa sehingga tugas akhir ini dapat selesai.
6. Seluruh teman-teman 2013 yang telah memberikan dukungan dan semangat.
7. Teman-teman b201crew Laboratorium yang telah memberikan suasana dan hiburan dalam pengerjaan tugas akhir ini.
8. Teman-teman b401 Laboratorium yang telah terbuka dan mengizinkan saya untuk mengerjakan tugas akhir di b401.
9. Semua pihak yang telah membantu saya dengan bersedia menjadi objek pengujian tugas akhir saya.

Penulis menyadari bahwa penulisan buku tugas akhir ini memiliki banyak kekurangan, oleh sebab itu penulis mengharapkan kritik dan saran yang membangun dari pembaca. Penulis berharap semoga tugas akhir ini dapat memberikan manfaat bagi kita semua.

Surabaya, Desember 2017

Penulis



# DAFTAR ISI

<b>Abstrak</b>	<b>i</b>
<b>Abstract</b>	<b>iii</b>
<b>KATA PENGANTAR</b>	<b>v</b>
<b>DAFTAR ISI</b>	<b>vii</b>
<b>DAFTAR GAMBAR</b>	<b>xi</b>
<b>DAFTAR TABEL</b>	<b>xv</b>
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Perumusan Masalah . . . . .	2
1.3 Tujuan . . . . .	2
1.4 Batasan Masalah . . . . .	2
1.5 Sistematika Penulisan . . . . .	3
<b>2 TINJAUAN PUSTAKA</b>	<b>5</b>
2.1 Robot Pelayan ( <i>Service Robot</i> ) . . . . .	5
2.2 <i>Turtlebot</i> . . . . .	6
2.2.1 <i>iCLebo Kobuki Base</i> . . . . .	7
2.2.2 <i>RGBD Sensor</i> . . . . .	10
2.2.3 <i>Pemroses Unit</i> . . . . .	15
2.3 Deteksi Manusia . . . . .	16
2.3.1 <i>Histogram of Oriented Gradient</i> (HOG) . . . .	16
2.3.2 <i>Skeletal Tracking</i> . . . . .	18
<b>3 DESAIN SISTEM</b>	<b>21</b>
3.1 Desain Sistem . . . . .	21
3.2 Desain Robot . . . . .	22
3.3 Desain Metode Object Following . . . . .	24
3.3.1 <i>Object Following</i> menggunakan <i>Histogram of Oriented Gradient</i> (HOG) . . . . .	24

3.3.2	<i>Object Following</i> menggunakan <i>Skeletal Tracking</i> . . . . .	27
<b>4</b>	<b>Implementasi</b>	<b>31</b>
4.1	Implementasi Robot . . . . .	31
4.2	Implementasi Metode <i>Object Following</i> . . . . .	33
4.2.1	Implementasi Deteksi dengan metode HOG . . . . .	33
4.2.2	Implementasi pada metode <i>Skeletal Tracking</i> . . . . .	36
<b>5</b>	<b>PENGUJIAN DAN ANALISIS</b>	<b>39</b>
5.1	Mekanisme Pengujian . . . . .	39
5.1.1	Pengujian Robot . . . . .	39
5.1.2	Pengujian Metode <i>Histogram of Oriented Gradient</i> (HOG) . . . . .	39
5.1.3	Pengujian Metode <i>Skeletal Tracking</i> . . . . .	40
5.1.4	Pengujian Metode HOG dan <i>Skeletal Tracking</i> pada berbagai Pose Tubuh . . . . .	40
5.1.5	Pengujian Kemampuan Robot Melakukan Penjajakan . . . . .	40
5.1.6	Pengujian Penjajakan ketika terdapat bayangan dari objek . . . . .	40
5.1.7	Pengujian Penjajakan ketika Objek Berjalan Menggunakan Alat Bantu . . . . .	41
5.1.8	Pengujian Penjajakan Diluar Ruangan ( <i>Outdoor</i> ) . . . . .	41
5.2	Hasil dan Pembahasan . . . . .	41
5.2.1	Pengujian Robot . . . . .	41
5.2.2	Pengujian Metode <i>Histogram of Oriented Gradient</i> (HOG) . . . . .	43
5.2.3	Pengujian Metode <i>Skeletal Tracking</i> . . . . .	46
5.2.4	Metode HOG dan <i>skeletal tracking</i> pada Berbagai Posisi Tubuh . . . . .	48
5.2.5	Pengujian Penjajakan oleh Robot . . . . .	55
5.2.6	Pengujian Penjajakan ketika Terdapat Bayangan dari Objek . . . . .	65
5.2.7	Pengujian Penjajakan ketika Objek Berjalan Menggunakan Alat Bantu . . . . .	66

5.2.8	Pengujian Penjajakan Diluar Ruangan (Outdoor) . . . . .	67
<b>6</b>	<b>PENUTUP</b>	<b>69</b>
6.1	Kesimpulan . . . . .	69
6.2	Saran . . . . .	69
	<b>DAFTAR PUSTAKA</b>	<b>71</b>
	<b>Biografi Penulis</b>	<b>75</b>

*Halaman ini sengaja dikosongkan*

# DAFTAR GAMBAR

2.1	Robot Buddy (kiri) dan Robot Pepper (kanan) [1] [2]	5
2.2	Turtlebot . . . . .	6
2.3	<i>iClebo kobuki Base</i> [3] . . . . .	7
2.4	<i>Docking Charger Station</i> [4] . . . . .	8
2.5	Panel Kontrol Kobuki [5] . . . . .	9
2.6	konfigurasi menggunakan serial port [5] . . . . .	9
2.7	Data sheet pin port serial [5] . . . . .	10
2.8	<i>Kinect Xbox 360</i> . . . . .	11
2.9	Area sensor Kinect [6] . . . . .	12
2.10	<i>Asus Xtion Pro</i> . . . . .	13
2.11	Hubungan antara kedalaman relatif dan perbedaan yang diukur. . . . .	14
2.12	<i>intel NUC</i> . . . . .	16
2.13	Deteksi Manusia dengan metode HOG [7] . . . . .	17
2.14	Skeletal Tracking . . . . .	18
2.15	Diagram OpenNi dan NiTE [8] . . . . .	19
2.16	Posisi-posisi Joint . . . . .	20
3.1	Diagram Sistem Robot . . . . .	21
3.2	Desain Robot . . . . .	22
3.3	Tata letak perangkat pada robot . . . . .	23
3.4	Desain Plate pada Robot . . . . .	23
3.5	Blok Diagram Penelitian menggunakan HOG . . . . .	24
3.6	Alur Deteksi Manusia . . . . .	25
3.7	Proses Penentuan Posisi Objek . . . . .	26
3.8	Pembagian Daerah Posisi Objek . . . . .	26
3.9	Blok Diagram Penelitian menggunakan <i>skeletal tracking</i> . . . . .	28
3.10	Proses deteksi manusia pada <i>skeletal tracking</i> . . . . .	29
3.11	Jarak Robot dengan objek . . . . .	29
4.1	Bentuk Robot . . . . .	31
4.2	Tata letak perangkat pada robot . . . . .	32
4.3	Desain Plate pada Robot . . . . .	32
4.4	Citra RGB dan citra <i>depth</i> . . . . .	33

4.5	Hasil deteksi menggunakan HOG . . . . .	34
4.6	Hasil objek pada posisi 1 . . . . .	34
4.7	Hasil objek pada posisi 2 . . . . .	35
4.8	Hasil ketika objek jauh . . . . .	35
4.9	Hasil ketika objek berada di posisi ideal . . . . .	36
4.10	Hasil ketika objek terlalu dekat . . . . .	36
4.11	Hasil tangkapan sensor depth . . . . .	37
4.12	Proses kalibrasi pada objek tersebut . . . . .	37
4.13	Proses tracking objek . . . . .	38
4.14	Hasil Skeletal tracking dengan nilai beberapa joint pada sumbu x,y,z . . . . .	38
5.1	Peta percobaan pengisian daya otomatis . . . . .	42
5.2	gambar a proses deteksi pada resolusi 640x480, gam- bar b proses deteksi pada 480x320, gambar c proses deteksi pada resolusi 320x240 . . . . .	44
5.3	Macam-macam Posisi yang akan diuji . . . . .	48
5.4	Hasil Deteksi HOG yang kurang tepat . . . . .	51
5.5	Hasil deteksi menggunakan <i>skeletal tracking</i> . . . . .	54
5.6	Skema pengujian robot dalam mendeteksi objek. Gam- bar (a) mendeteksi objek dari depan, (b) mendeteksi objek dari samping, (c) mendeteksi objek dari belakang . . . . .	56
5.7	Gambar (a) hasil deteksi dari depan, (b) hasil deteksi dari samping, (c) hasil deteksi dari belakang . . . . .	57
5.8	Skema pengujian kemampuan penjajakan objek oleh robot <i>skeletal tracking</i> . . . . .	57
5.9	Jalur pergerakan robot selama melakukan penjajakan terhadap objek . . . . .	58
5.10	Jalur robot selama melakukan penjajakan dengan ja- lur zigzag . . . . .	59
5.11	Hasil penjajakan dengan pencahayaan normal . . . . .	59
5.12	Hasil penjajakan dengan pencahayaan gelap . . . . .	60
5.13	Ilustrasi pengujian kecepatan robot . . . . .	60
5.14	Grafik jarak objek dengan robot terhadap waktu . . . . .	61
5.15	grafik kecepatan robot terhadap waktu . . . . .	61
5.16	Skema pengujian 2 orang . . . . .	62
5.17	Hasil deteksi menggunakan <i>skeletal tracking</i> pada sa- at terdapat 2 orang . . . . .	63



5.18 Hasil pengujian saat terdapat 2 orang . . . . .	64
5.19 Ilustrasi Pengujian . . . . .	65
5.20 Hasil yang didapat robot, A hasil citra RGB dan B hasil deteksi manusia dengan <i>depth</i> . . . . .	66
5.21 Ilustrasi Pengujian . . . . .	66
5.22 proses deteksi ketika menggunakan alat bantu . . . .	67
5.23 Hasil deteksi diluar ruangan . . . . .	68

*Halaman ini sengaja dikosongkan*

## DAFTAR TABEL

2.1	Bagian-bagian Turtlebot . . . . .	7
2.2	Fitur Sensor Kinect . . . . .	12
2.3	Spesifikasi Asus Xtion Pro [9] . . . . .	13
2.4	Perbedaan SDK Kinect . . . . .	19
5.1	Hasil Pengukuran pergerakan robot . . . . .	41
5.2	Hasil Pengukuran pergerakan robot . . . . .	43
5.3	Waktu dalam memproses tiap Frame . . . . .	44
5.4	Hasil pendeteksian HOG pada beberapa macam jarak (bagian 1) . . . . .	45
5.5	Hasil pendeteksian HOG pada beberapa macam jarak (bagian 2) . . . . .	46
5.6	Hasil pendeteksian <i>skeletal tracking</i> terhadap objek pada jarak tertentu (bagian 1) . . . . .	47
5.7	Hasil pendeteksian <i>skeletal tracking</i> terhadap objek pada jarak tertentu (bagian 2) . . . . .	48
5.8	Hasil Deteksi dengan menggunakan metode HOG (bagian 1) . . . . .	49
5.9	Hasil Deteksi dengan menggunakan metode HOG (bagian 2) . . . . .	50
5.10	Hasil Deteksi dengan menggunakan metode HOG (bagian 3) . . . . .	51
5.11	Hasil Deteksi dengan menggunakan metode <i>Skeletal Tracking</i> (bagian 1) . . . . .	52
5.12	Hasil Deteksi dengan menggunakan metode <i>Skeletal Tracking</i> (bagian 2) . . . . .	53
5.13	Hasil Deteksi dengan menggunakan metode <i>Skeletal Tracking</i> (bagian 3) . . . . .	54
5.14	Waktu dalam proses deteksi objek . . . . .	56

*Halaman ini sengaja dikosongkan*

# **BAB 1**

## **PENDAHULUAN**

Pada bab ini dijelaskan mengenai latar belakang permasalahan yang menjadi acuan dalam pengerjaan tugas akhir, perumusan dan batasan masalah yang dijawab melalui luaran tugas akhir berdasarkan tujuan yang ingin dicapai dan terdapat pula penjelasan terkait sistematika penulisan dari laporan tugas akhir ini.

### **1.1 Latar Belakang**

Seiring dengan kemajuan teknologi dan ilmu pengetahuan telah memicu timbulnya berbagai perubahan dalam masyarakat, dengan meningkatkan angka umur harapan hidup (UHH). Namun peningkatan UHH ini dapat mengakibatkan terjadinya perubahan struktur demografi yaitu peningkatan populasi lanjut usia (lansia). Berdasarkan laporan Perserikatan Bangsa-Bangsa 2011, pada tahun 2000-2005 UHH adalah 66,4 tahun (dengan persentase populasi lansia tahun 2000 adalah 7,74%), angka ini akan meningkat pada tahun 2045-2050 yang diperkirakan UHH menjadi 77,6 tahun (dengan persentase populasi lansia tahun 2045 adalah 28,68%). Begitu pula dengan laporan Badan Pusat Statistik (BPS) di Indonesia terjadi peningkatan UHH. Pada tahun 2000 UHH di Indonesia adalah 64,5 tahun (dengan persentase populasi lansia adalah 7,18%). Angka ini meningkat menjadi 69,43 tahun pada tahun 2010 (dengan persentase populasi lansia adalah 7,56%) dan pada tahun 2011 menjadi 69,65 tahun (dengan persentase populasi lansia adalah 7,58%) [10].

Dengan bertambahnya umur, terjadi proses penuaan akan berdampak pada aspek kehidupan baik sosial, ekonomi, maupun kesehatan. Dari aspek kesehatan fungsi fisiologis mengalami penurunan akibat proses degeneratif (penuaan) sehingga penyakit tidak menular yang banyak muncul pada usia lanjut. Selain itu masalah degeneratif menurunkan daya tahan tubuh sehingga rentan terkena infeksi penyakit menular. Penyakit tidak menular pada lansia di antaranya hipertensi, stroke, diabetes mellitus dan radang sendi atau rematik. Sedangkan penyakit menular yang diderita adalah

tuberkulosis, diare, pneumonia dan hepatitis [11].

Hal ini membuat kondisi lansia perlu dipantau setiap saat, mulai dari kesehatan mereka, pola kehidupan, dan aktifitas mereka. Sehingga sewaktu-waktu terjadi sesuatu terhadap kondisi lansia tersebut keluarga atau petugas medis dapat dengan cepat melakukan tindakan pertolongan terhadap lansia tersebut. Namun manusia memiliki keterbatasan dan aktifitas sendiri sehingga pemantauan tidak dapat dilakukan terus-menerus. Untuk itu diperlukan sebuah robot service yang dapat memantau kondisi lansia setiap saat. Robot service tersebut memiliki fungsi untuk memantau kondisi vital, pengingat waktu minum obat, mampu mengikuti lansia kemanapun pergi agar dapat memantau aktifitas atau kegiatan lansia tersebut dan dapat mengirimkan pesan darurat kepada keluarga atau pelayanan kesehatan setempat ketika terjadi masalah dengan kondisi lansia tersebut.

## 1.2 Perumusan Masalah

Diperlukannya sistem yang dapat digunakan pada *robot service* agar dapat mengikuti kemanapun lansia bergerak dan melakukan pemantauan terhadap kondisi dan aktifitas lansia.

## 1.3 Tujuan

Tujuan dari tugas akhir ini adalah mengimplementasikan metode *object following* pada *robot service* sehingga robot dapat bergerak mengikuti pergerakan objek dalam hal ini adalah manusia manusia.

## 1.4 Batasan Masalah

Batasan masalah pada tugas akhir ini adalah :

1. Pengujian berada di dalam ruangan persegi tanpa adanya halangan.
2. Pengujian hanya dilakukan di ruangan yang berisi satu orang.
3. Robot service yang digunakan adalah turtlebot 2.
4. Sistem beroperasi di dalam ruangan (*indoor*)



## 1.5 Sistematika Penulisan

Laporan tugas akhir ini tersusun dalam sistematika dan terstruktur sehingga mudah dipahami dan dipelajari oleh pembaca maupun seseorang yang ingin melanjutkan tugas akhir ini. Alur sistematika penulisan laporan tugas akhir ini yaitu :

1. BAB I Pendahuluan

Bab ini berisi uraian tentang latar belakang permasalahan, batasan masalah yang diteliti, tujuan penelitian, dan sistematika laporan.

2. BAB II Dasar Teori

Pada bab ini berisi tentang uraian sistematis teori penunjang yang menjadi referensi dalam pengerjaan tugas akhir. Teori penunjang tersebut meliputi *robot service*, *Turtlebot* dan Deteksi Manusia.

3. BAB III Desain Sistem

Bab ini berisi penjelasan tentang desain atau perancangan sistem yang diwujudkan dalam bentuk blok diagram penelitian

4. BAB IV Implementasi Sistem

Pada bab ini berisi tentang hasil yang telah dibuat berdasarkan desain sistem pada bab sebelumnya

5. BAB V Pengujian dan Analisis

Bab ini menjelaskan mengenai hasil pengujian sistem *object following* pada *robot service* dan analisis terkait pengujian baik secara keseluruhan maupun secara khusus terhadap hasil *object following* dalam pengujian.

6. BAB VI Penutup

Bab ini merupakan penutup yang berisi kesimpulan yang diambil dari tugas akhir dan pengujian yang telah dilakukan. Saran dan kritik yang membangun untuk mengembangkan lebih lanjut juga dituliskan pada bab ini.

*Halaman ini sengaja dikosongkan*

## BAB 2

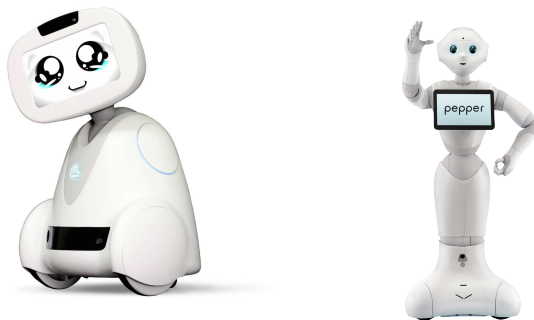
### TINJAUAN PUSTAKA

Pada bab ini dijelaskan mengenai beberapa teori penunjang yang menjadi referensi dalam pengerjaan tugas akhir. Teori penunjang tersebut meliputi robot service, *Turtlebot* dan Deteksi Manusia.

#### 2.1 Robot Pelayan (*Service Robot*)

Saat ini robot hampir digunakan dalam berbagai kehidupan manusia. Seperti robot pelayan pada cafe, hotel dan tempat makan. Mereka menggunakan robot untuk mengantar barang atau pesan-an kepada pelanggan. Terdapat juga robot yang digunakan dalam bidang kesehatan seperti robot untuk pengantar obat dan makanan di rumah sakit. Bahkan terdapat beberapa robot yang dapat berinteraksi dengan manusia.

Robot pelayan merupakan robot yang difungsikan untuk membantu manusia. Biasanya robot ini digunakan untuk melakukan pekerjaan yang berulang-ulang, berbahaya, kotor, jauh, dan pekerjaan rumah tangga. Robot tersebut biasanya bekerja secara otomatis atau dioperasikan oleh sistem tertentu. Menurut Federasi Robotika Internasional (IFR) pengertian *service robot* adalah robot yang beroperasi secara *semiotomatis* atau sepenuhnya mandiri untuk melakukan layanan yang berguna bagi kesejahteraan manusia [12].

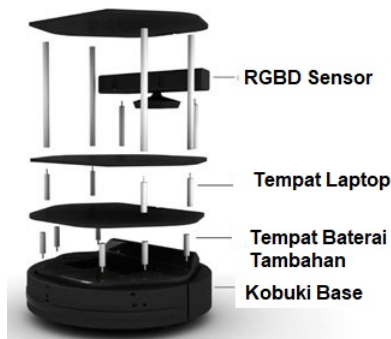


**Gambar 2.1:** Robot Buddy (kiri) dan Robot Pepper (kanan) [1] [2]

## 2.2 *Turtlebot*

Saat ini robot tidak hanya digunakan dalam bidang industri saja, namun saat ini robot telah menjadi hal umum dalam kehidupan sehari-hari. Pendidikan, pertanian, perikanan, militer, serta beberapa kegiatan yang mempunyai resiko yang sangat besar telah menggunakan tenaga robot agar lebih efisien dan cepat. Salah satu jenis robot yang dapat digunakan untuk membantu manusia adalah *turtlebot*.

*Turtlebot* merupakan sebuah *robot service open platform* yang dapat digunakan untuk pembelajaran serta penelitian tentang robot. Desain *turtlebot* dapat diubah-ubah sesuai kebutuhan sehingga robot ini tidak terbatas untuk penggunaan tertentu. *Turtlebot* terdiri dari kobuki base, *RGBD sensor*, netbook atau *Single Board Processor* (SBC), dan rangka yang digunakan sebagaiudukan komponen lainnya. Gambar 2.2 merupakan tampilan dari *turtlebot*.

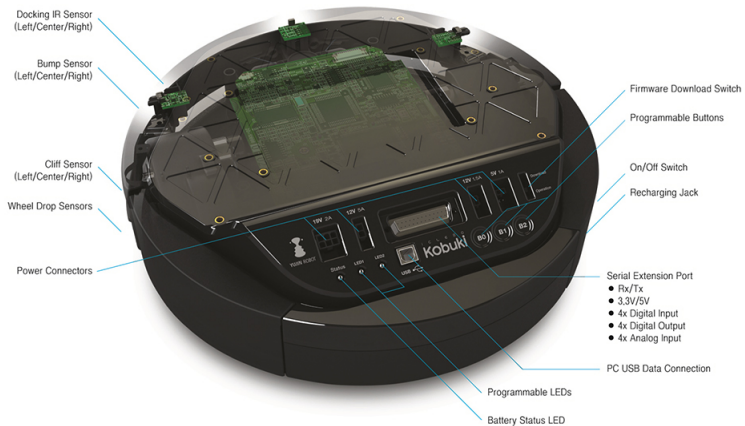


Gambar 2.2: *Turtlebot*

**Tabel 2.1:** Bagian-bagian Turtlebot

No	Nama bagian	Fungsi
1	iClebo Kobuki Base	Base dasar dan penggerak pada turtlebot
2	RGBD Sensor	sebagai sensor pada turtlebot
3	Unit Pemroses	Pemroses semua data pada turtlebot dan sebagai kontrol pada turtlebot

### 2.2.1 *iClebo Kobuki Base*



**Gambar 2.3:** *iClebo kobuki Base*[3]

*Iclebo Kobuki* merupakan *low-cost mobile research base* yang digunakan untuk pendidikan dan penelitian dalam bidang robotika. Pada *iClebo kobuki* memiliki terdapat sensor, aktuator, *Battery Lithium-Ion*, pin konektor, lampu indikator, dan 3 buah tombol. Sensor yang terdapat pada kobuki berupa 3 buah bumper sensor, 3 buah cliff sensor, 3 buah docking IR sensor, sensor gyro, dan 2 buah *wheel drop sensor* [13]. Sensor bumper, cliff, dan docking IR terletak pada sisi kiri, kanan dan depan, sedangkan *wheel drop sensor* terletak pada 2 buah roda penggerak. Sensor bumper berfungsi

sebagai pendeteksi adanya benturan pada kobuki, sensor cliff berfungsi untuk membaca ketinggian posisi kobuki terhadap alas atau pijakan kobuki. Docking IR sensor berfungsi untuk mencari posisi *docking charger* sehingga sewaktu baterai pada kobuki base meleleh kobuki dapat mengisi baterai dengan otomatis.



**Gambar 2.4:** *Docking Charger Station* [4]

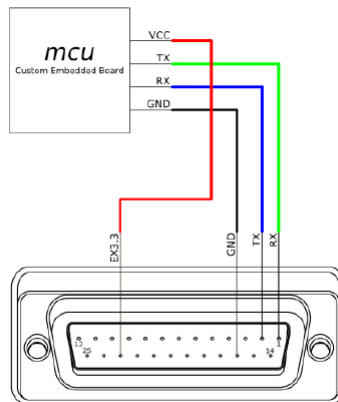
Pada bagian belakang kobuki terdapat panel yang berisi lampu 3 buah lampu indikator, buah *touch button*, 4 jenis konektor daya, konektor serial, konektor usb type B, dan sebuah *switch button*. Lampu indikator dan *touch button* pada kobuki dapat diprogram sesuai kebutuhan namun hanya 2 buah indikator yang dapat diprogram sedangkan 1 buah indikator tidak dapat digunakan karena telah digunakan sebagai indikator baterai. Kobuki dapat memberikan daya pada perangkat lain melalui konektor-konektor yang telah disediakan. Daya yang disiapkan kobuki untuk menyuplai perangkat lain terbagi kedalam 4 macam 19V 2A, 12V 5A, 12V 1.5A, dan 5V 1A. *switch button* pada kobuki berfungsi menentukan fungsi dari kobuki, terdapat 2 mode fungsi yaitu *download* dan *operation*. *Download mode* digunakan ketika mengunduh firmware kedalam perangkat kobuki sedangkan *operation mode* digunakan saat kobuki melakukan fungsi kerja sesuai sistem.



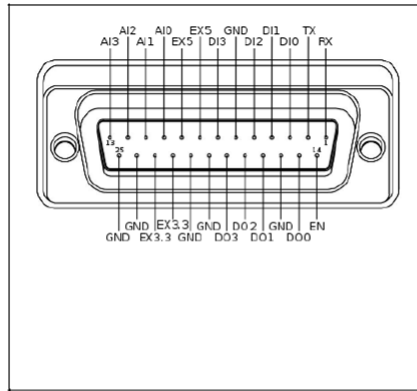


**Gambar 2.5:** Panel Kontrol Kobuki [5]

Kobuki tidak dapat beroperasi tanpa adanya sebuah proses unit (SBC atau netbook). Kobuki hanya bisa memberikan data dari sensor-sensor yang terpasang kemudian diteruskan menuju proses unit dan melakukan tindakan sesuai dengan hasil proses. Komunikasi data antara kobuki dan SBC ata netbook dapat menggunakan port usb atau port serial. Jika pada menggunakan port serial konfigurasi ditunjukkan oleh gambar 2.6. Pada port serial selain menggunakan serial data untuk melakukan transfer data, juga terdapat pin untuk digital input/output dan pin analog input/output (gambar 2.7).



**Gambar 2.6:** konfigurasi menggunakan serial port [5]



Keterangan :

- Rx/Tx : Serial Data (Tegangan 3.3 V)
- EX3.3 dan EX5 : Supply Daya 3.3V 1A dan 5V 1A
- DI0~3 : Digital Input (High: 3.3V – 5V, Low : 0V)
- DO0~3 : Digital Output
- AI0~3 : Analog Input (12 bit ADC, 0 - 4096, 0 ~ 3.3V)
- GND : Ground
- EN : Untuk Boud Tambahan

**Gambar 2.7:** Data sheet pin port serial [5]

Pada Turtlebot *iCleo kobuki* digunakan sebagai penggerak dan base dasar pada rangka *turtlebot*. Kobuki dapat mengangkat beban maksimal 5 Kg pada medan lantai dan maksimal 4 kg pada medan karpet . Suplai daya pada kobuki menggunakan *battery Lithium-Ion* 14.8 V 2200 mAh (*small*) atau 4400 mAh (*Large*). Dengan daya tersebut kobuki dapat beroperasi selama 3 jam dengan baterai 2200 mAh dan 7 jam pada baterai 4400 mAh. Proses pengisian daya pada kobuki memerlukan waktu 2 jam untuk baterai 2200 mAh dan 3 jam untuk baterai 4400 mAh [5].

### 2.2.2 RGBD Sensor

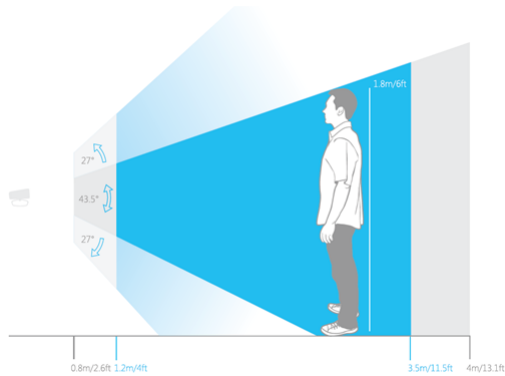
Turtlebot juga dapat dilengkapi dengan *RGBD sensor* yang berfungsi sebagai sensor tambahan pada turtlebot. Terdapat 2 jenis

*RGBD sensor* yang dapat digunakan pada turtlebot yaitu Kinect dan Asus Xtion Pro. Kinect merupakan salah satu produk dari *Microsoft* untuk Xbox 360 dan Game console, serta dapat digunakan juga untuk personal komputer. Teknologi yang terdapat pada Kinect dikembangkan oleh perusahaan *developer* teknologi kamera bernama *PrimeSense*. *PrimeSense* mengembangkan sistem yang dapat menginterpretasikan gestur secara spesifik, sehingga kontrol secara *hands-free* dapat dilakukan pada perangkat elektronik menggunakan proyektor infrared dan kamera, serta *microchip* khusus untuk melacak pergerakan obyek dan individu pada bidang tiga dimensi. Pada kinect terdapat *RGB sensor*, *depth Sensor*, *Stereo microphone*, dan motor sebagai pengatur posisi sudut kamera [14].



**Gambar 2.8:** *Kinect Xbox 360*

Sensor kedalaman terdiri atas kombinasi *infrared laser projector* dan *monochrome CMOS sensor*. *infrared laser projector* berfungsi dalam mentransmisi cahaya *invisible near-infrared* ke seluruh bagian ruangan yang terjangkau oleh sensor Kinect. Jarak yang harus dipenuhi ketika sensor Kinect digunakan adalah 1,2 meter hingga 3,5 meter, apabila jarak yang digunakan melebihi atau kurang dari jarak yang telah ditetapkan maka penangkapan gerak dari objek tidak akan terjadi atau hanya tertangkap sebagian. Fitur *array microphone* berfungsi sebagai masukan suara, dapat digunakan memerintah program tertentu melalui pengolahan program komputer, dapat juga dikembangkan untuk mengenal suara manusia. Agar dapat mengatur jangkauan tangkapan dari Kinect, kita dapat merubah posisi dari *motorized tilt* yang terdapat pada Kinect. Pada gambar 2.9 menunjukkan sudut dan jarak dari Sensor Kinect [15].



**Gambar 2.9:** Area sensor Kinect [6]

Agar lebih jelas detail dari spesifikasi sensor Kinect terdapat pada tabel 2.2 .

**Tabel 2.2:** Fitur Sensor Kinect

<i>Sensor</i>	<ol style="list-style-type: none"> <li>1. <i>color and Depth sensing Lenses</i></li> <li>2. <i>Voice microphone array</i></li> <li>3. <i>motorized tilt</i></li> </ol>
<i>Field Of View</i>	<ol style="list-style-type: none"> <li>1. <i>Horizontal field of view : 57 degree</i></li> <li>2. <i>Vertical field of view : 43 degree</i></li> <li>3. <i>Physical tilt range : <math>\pm 27</math> degree</i></li> <li>4. <i>Depth sensor : 1,2 m - 3,5 m</i></li> </ol>
<i>Data Stream</i>	<ol style="list-style-type: none"> <li>1. <i>320x240 16 bit depth @ 30 frames/sec</i></li> <li>2. <i>640x480 32 bit color @ 30 frames/sec</i></li> <li>3. <i>16 bit audio @ 16khz</i></li> </ol>
<i>Skeleton tracking</i>	<ol style="list-style-type: none"> <li>1 <i>Tracks up to 6 people, including 2 active player</i></li> <li>2. <i>Tracks 20 joints per active player</i></li> </ol>
<i>Audio sistem</i>	<ol style="list-style-type: none"> <li>1 <i>Speech recognition in multiple</i></li> </ol>

Asus Xtion Pro merupakan produk sensor RGBD dari Asus. Asus Xtion Pro memiliki *RGB sensor*, *RGB sensor*, dan *microphone*. Berbeda dengan Kinect pada Asus Xtion Pro tidak terdapat power supply tambahan karena tidak dilengkapi dengan motor se-

hingga daya dari port USB cukup sebagai daya pada sensor ini. Asus Xtion Pro memiliki 2 macam ukuran citra, ukuran pertama 320x240 dengan 60 fps dan ukuran kedua 640x480 dengan 30 fps. Spesifikasi lebih jelas tentang Asus Xtion Pro terdapat pada tabel 2.3. Kedua sensor tersebut memiliki kemampuan untuk mendeteksi gerakan tangan, mengetahui kedalaman obyek, deteksi wajah, deteksi tubuh dan perintah menggunakan suara [16].



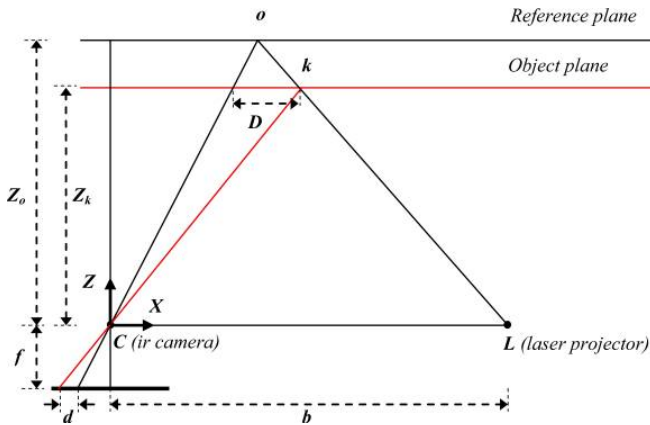
**Gambar 2.10:** *Asus Xtion Pro*

**Tabel 2.3:** Spesifikasi Asus Xtion Pro [9]

<i>Sensor</i>	<i>Depth</i>
<i>Depth Image Size</i>	VGA (640x480 @ 30 fps) QVGA (320x240 @ 60 fps)
<i>Field Of View</i>	58° H, 45° V, 70° D (Horizontal, Vertical, Diagonal)
<i>Distance of Use</i>	between 0,8m - 3,5m
<i>Power Comsumtion</i>	Below 2,5 W
<i>interface</i>	USB 2.0
<i>Platform</i>	Intel X86 and AMD
<i>OS Support</i>	Win 32/64:XP/Vista/7/8 Linux Ubuntu 10.10 32/64bit
<i>Software</i>	Software Development Kit (OpenNi SDK bundled)
<i>Programming Langue</i>	C++/C# (windows) c++ (Linux) JAVA
<i>Dimension</i>	18 x 3,5 x 5

Pada sensor *depth* untuk mengukur koordinat objek dan kedalaman/jarak objek dapat menggunakan metode triangulasi. Gambar 2.11 mengilustrasikan hubungan antara jarak titik objek k ke

sensor relatif terhadap bidang referensi dan perbedaan terukur ( $d$ ). Untuk mengekspresikan koordinat 3D dari titik-titik objek kita mempertimbangkan sistem koordinat kedalam dengan asal-usulnya di pusat perspektif kamera inframerah. Sumbu Z pada bidang gambar ke arah objek, sumbu X tegak lurus terhadap sumbu Z ke arah garis dasar  $b$  antara pusat kamera inframerah dan proyektor laser, dan sumbu Y ortogonal ke X dan Z membuat hak menyerahkan sistem koordinat.



**Gambar 2.11:** Hubungan antara kedalaman relatif dan perbedaan yang diukur.

Asumsikan bahwa benda berada pada bidang referensi pada jarak  $Z_o$  ke sensor, dan sebuah spekel pada objek ditangkap pada bidang gambar kamera inframerah. Jika objek digeser mendekati (atau lebih jauh dari) sensor lokasi bintang pada bidang gambar akan dipindahkan ke arah X. Ini diukur dalam ruang gambar sebagai disparitas  $d$  sesuai dengan titik  $k$  di ruang objek. Berdasarkan gambar 2.11 dapat dibuat persamaan 2.1 dan 2.2. Dimana  $Z_k$  menunjukkan jarak (kedalaman) titik  $k$  pada ruang objek,  $b$  adalah panjang dasar,  $f$  adalah focal length dari kamera inframerah,  $D$  adalah perpindahan dari titik  $k$  pada ruang objek, dan  $d$  adalah disparitas yang diamati. Berdasarkan persamaan 2.1 dan 2.2, dilakukan substitusi terhadap  $D$  pada persamaan 2.2 dengan nilai  $D$  pada persamaan 2.1 sehingga

mendapat persamaan 2.3.

$$\frac{D}{b} = \frac{Z_o - Zk}{Z_o} \quad (2.1)$$

$$\frac{d}{f} = \frac{D}{Zk} \quad (2.2)$$

$$Zk = \frac{Z_o}{1 + \frac{Z_o}{fb} * d} \quad (2.3)$$

### 2.2.3 Pemroses Unit

Turtlebot memerlukan sebuah perangkat proses unit agar robot dapat bekerja. Beberapa proses unit dapat digunakan pada turtlebot seperti notebook, *Next Unit of Computing*(NUC), *single board computer*, dan *embedded board*. Jika menggunakan SBC atau *embedded board* sebagai unit pemroses, komunikasi dengan kobuki melalui port serial. Namun bila menggunakan notebook atau NUC menggunakan port usb yang tersedia pada kobuki, selain itu diperlukan sebuah driver atau framework agar system dapat berinteraksi dengan perangkat. *framework Robot Operating System (ROS)* salah satu yang dapat digunakan sebagai penyambung dengan kobuki. ROS berisi beragam peralatan (*tools*), *library* dan driver yang bertujuan untuk memudahkan pembuatan program pada berbagai platform robot. Dengan *framework ROS* para pengembang yang berbeda-beda dapat bekerja sama mengembangkan suatu karya [17]. Pada ROS terdapat *library* dan driver yang diperlukan untuk pergerakan *iCub Kobuki*, sehingga kita dapat dengan mudah untuk menerima, dan mengirim data serta mengontrol *kobuki base*.



**Gambar 2.12:** *intel NUC*

## 2.3 Deteksi Manusia

Proses pendeteksian manusia dapat menggunakan kamera dapat dilakukan dengan berbagai cara salah satunya menggunakan metode *Histogram of Oriented Gradient* (HOG). Data yang didapat dari kamera kemudian diproses oleh HOG agar dapat membedakan objek manusia dan bukan manusia. Selain itu pendeteksian manusia dapat menggunakan sensor kedalaman (*depth*). *depth sensor* memberikan data berupa perbedaan jarak benda-benda yang berada di depan sensor. Metode yang digunakan untuk mendeteksi manusia menggunakan data sensor *depth* dapat menggunakan *skeletal tracking*.

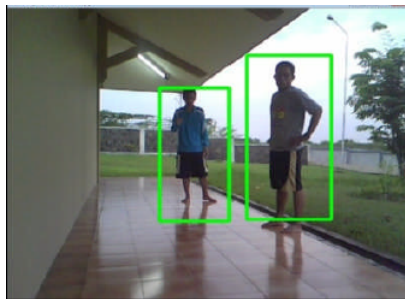
### 2.3.1 *Histogram of Oriented Gradient*(HOG)

*Histogram Oriented of Gradient* adalah deskripsi fitur yang digunakan dalam *computer vision* dan pemrosesan gambar untuk tujuan deteksi objek. Metode ini mirip dengan *edge oriented histogram*, *scale-invariant feature transform descriptor*, dan *shape contexts*, namun terdapat perbedaan dalam hal perhitungan pada grid dan menggunakan normalisasi kontras lokal yang *overlapping* untuk akurasi yang lebih baik. HOG meluas setelah Navneet Dalal dan Bill Triggs dari *computer science and automation* Institut Nasional Perancis (INRIA), mempresentasikan karya mereka dalam Konferensi *Computer Vision and Pattern Recognition* (CVPR). Karya mereka adalah menggunakan HOG untuk mendeteksi pejalan kaki dalam citra statis.



Proses pendeteksian objek manusia melalui citra terdapat beberapa tahap antara lain, normalisasi gamma warna, perhitungan gradien (*compute gradient*), *weight vote into spatial and orientation cells*, *overlapping spatial block*, *collect HOG over detection windows*, dan linier SVM. Langkah pertama adalah normalisasi nilai gamma dan warna, namun menurut Dalal dan Bill proses ini dapat diabaikan karena tidak berpengaruh terhadap hasil pada proses berikutnya. Setelah itu dilakukan proses perhitungan gradient pada proses ini setiap pixel dilakukan perhitungan dengan kernel tertentu pada arah *vertical* dan *horizontal*.

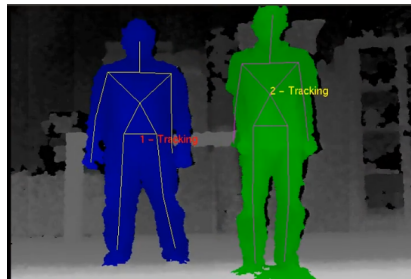
Selanjutnya dilakukan proses menghitung orientasi pada cell. Cell merupakan kumpulan pixel dengan ukuran dimensi tertentu (8x8). Pada tiap cell dilakukan proses agar dapat mengetahui orientasi / arah pada tiap cell. Cell-cell tadi kemudian di kelompokkan lagi menjadi sebuah blok. Tiap blok biasanya berukuran 2x2 cell. Blok-blok ini nanti digunakan untuk mempertajam nilai gradien pada suatu citra. Setelah itu dilakukan proses normalisasi pada blok dengan *overlap* sebesar 50 %. Hasil dari proses normalisasi berbentuk *HOG descriptor* dan akan digunakan sebagai masukan dari proses linier SVM. Hasil dari proses SVM menentukan pada citra tersebut terdapat bagian yang dideteksi sebagai manusia atau bukan [18].



**Gambar 2.13:** Deteksi Manusia dengan metode HOG [7]

### 2.3.2 *Skeletal Tracking*

*Skeletal Tracking* merupakan produk dari perusahaan *primesense*. *Primesense* adalah perusahaan penginderaan 3D yang berbasis di Tel-Aviv. Sebelumnya *primesense* bekerja dengan *microsoft* dalam membuat Kinect, suatu kontroler kamera dan sensor yang dapat mengenali gerakan penggunanya pada konsol game Xbox 360. Kinect juga mengeluarkan sebuah *software development kit* (SDK) yang dapat digunakan pengembangan aplikasi dengan menggunakan sensor Kinect di luar Xbox. Terdapat beberapa macam SDK yang dapat digunakan untuk Kinect diantaranya, Kinect for windows SDK, OpenKinect, dan OpenNI. Kinect untuk windows SDK dilengkapi dengan *driver* agar data dari sensor dapat diproses dan pelacakan gerakan manusia. Kinect SDK ini dirilis oleh *Microsoft* untuk pengembangan teknologi dengan aplikasi berbasis bahasa pemrograman *c++*, *c#* dan visual basic [15].



**Gambar 2.14:** Skeletal Tracking

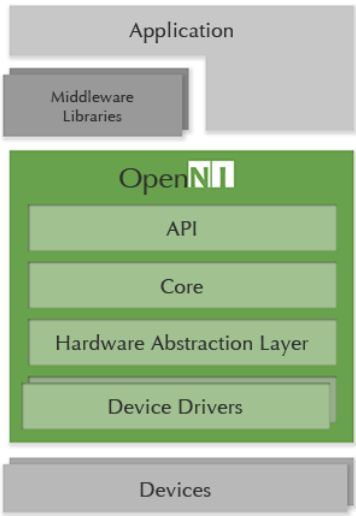
Selanjutnya *primesense* mengembangkan *middleware* yang dapat menganalisis data dari perangkat keras dan modul untuk OpenNi yang dapat memberikan *gesture* dan *skeleton tracking*. NiTE dapat mengidentifikasi pengguna dan melacak pergerakan mereka dan menyediakan *framework* API untuk menerapkan kontrol UI dengan *natural-interaction* berdasarkan isyarat [19]. Sedangkan OpenKinect merupakan komunitas terbuka dari orang-orang yang tertarik memanfaatkan perangkat Xbox Kinect dengan *personal computer* PC atau perangkat lain. OpenKinect terbuka secara umum dan memiliki library yang dapat digunakan pada *operating system* Windows, Linux dan Mac. Fokus utama OpenKinect adalah terciptanya

*software* libfreenect. Perbedaan lebih jelas mengenai SDK diatas terdapat pada tabel 2.4.

**Tabel 2.4:** Perbedaan SDK Kinect

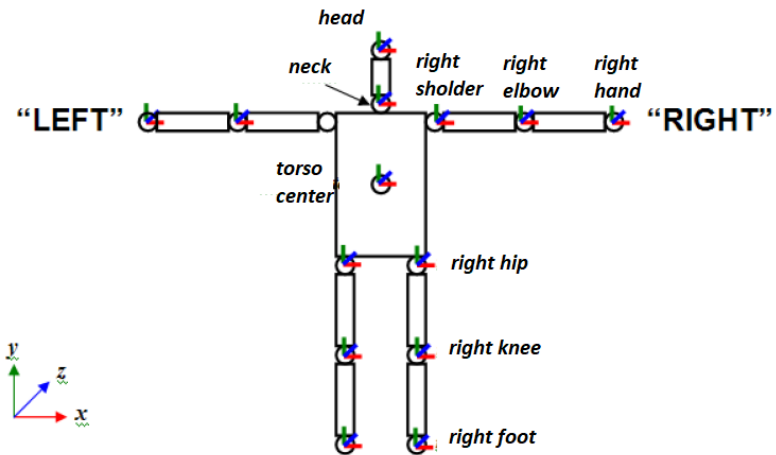
SDK	<i>Operating System</i>	<i>Skeleton mapping</i>	Kalibrasi
OpenKinect	Windows, Linux, OS X	Tidak Ada	Tidak Ada
OpenNI	Windows, Linux, OS X	Ada	Tidak Ada
Windows SDK	Windows	Ada	Ada

Deteksi tubuh manusia menggunakan *skeleton tracking*, data dari sensor diakses oleh OpenNi kemudian diproses oleh *middleware library*.



**Gambar 2.15:** Diagram OpenNi dan NiTE [8]

Pada OpenNi terdapat *device driver*, *hardware Abstraction Layer*, *Core*, dan *API*. *Device driver* berfungsi agar perangkat yang terhubung dapat melakukan transfer data atau data yang dikirim dari perangkat dapat diakses dan diterima oleh OpenNi. *Hardware Abstraction Layer* berfungsi agar OpenNi SDK dapat bekerja pada beberapa *device*. Hasil dari *middleware library* berupa data posisi joint dalam sumbu x, y, dan z. Terdapat 15 posisi joint yang dapat diambil datanya, posisi-posisi tersebut antara lain *head*, *neck*, *shoulder*, *elbow*, *hand*, *torso*, *hip*, *knee*, dan *foot* (gambar 2.16) [20]. Pada tiap posisi tersebut dapat diambil data berupa nilai jarak dari sumbu x, y, dan z. Selain itu 15 posisi tersebut digunakan dalam proses melukis bentuk bone pada objek.



**Gambar 2.16:** Posisi-posisi Joint

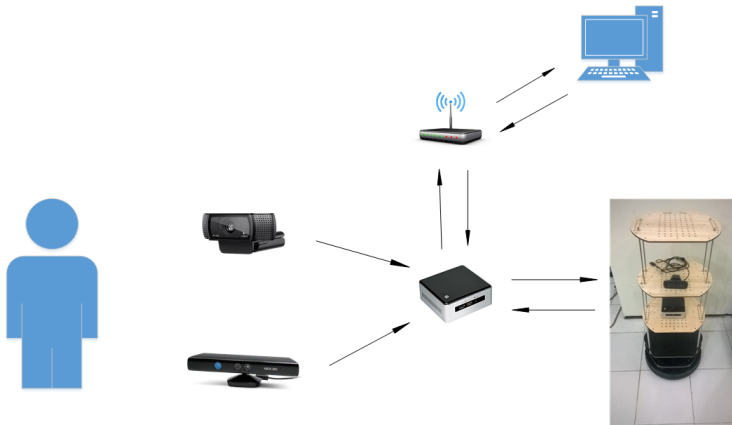
## BAB 3

# DESAIN SISTEM

Pada bab ini dijelaskan mengenai desain sistem yang diwujudkan dalam bentuk blok diagram penelitian dan implementasi sistem yang merupakan pelaksanaan teknis dari setiap blok diagram pada desain sistem.

### 3.1 Desain Sistem

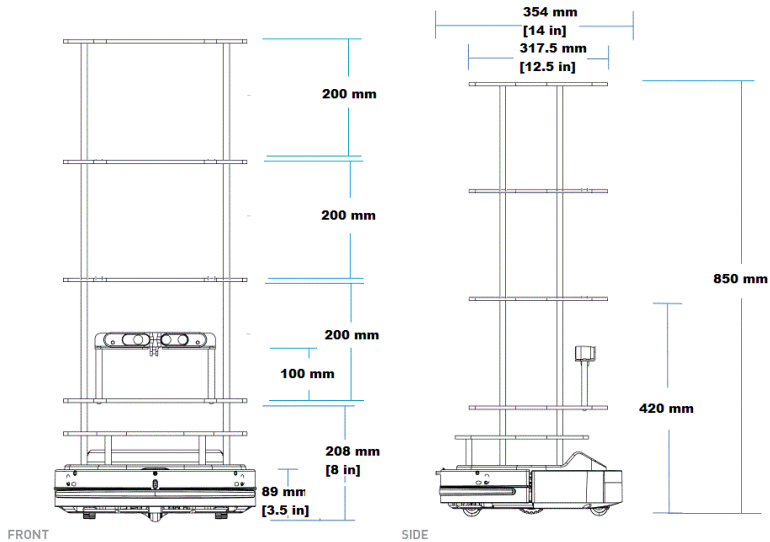
Desain sistem agar robot service dapat melakukan penjajakan terhadap suatu objek terlihat pada Gambar 3.1. Pada gambar tersebut robot service dipasang sebuah sensor RGBD dan NUC. Sensor RGBD ini berfungsi agar dapat menangkap benda yang ada di depan robot service. Selain itu sensor RGBD juga dapat memberikan besaran nilai jarak antara objek yang ada di depan robot. NUC pada sistem ini berfungsi sebagai unit pemroses dari citra yang ditangkap oleh sensor RGBD maupun data yang dikirim oleh kobuki base serta alat-alat yang terpasang pada dan juga sebagai pengelola gerakan robot. Sehingga robot dapat mengikuti objek (manusia) kemanapun gerakan objek tersebut.



**Gambar 3.1:** Diagram Sistem Robot

### 3.2 Desain Robot

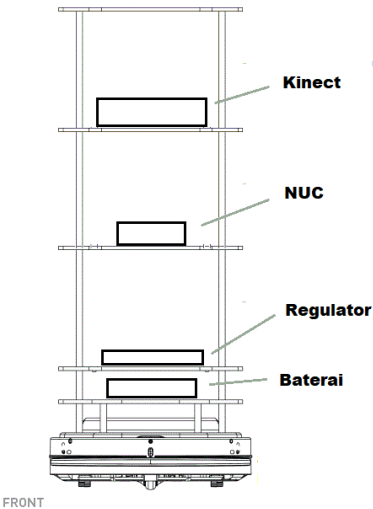
Robot yang digunakan pada tugas akhir ini adalah robot TurtleBot 2. TurtleBot yang semula memiliki 3 tingkat *plate* diubah menjadi 5 *plate*. *Plate* tersebut digunakan sebagai tempat peletakan benda-benda yang nanti dipergunakan dalam menjalankan sistem, seperti sensor RGBD, laptop atau NUC, dan sebagai tempat bagi baterai tambahan. Robot ini memiliki ketinggian 850 mm dan diameter bentuk 354 mm. Dengan bentuk seperti itu diharapkan robot dapat bergerak dengan mudah dan fleksibel.



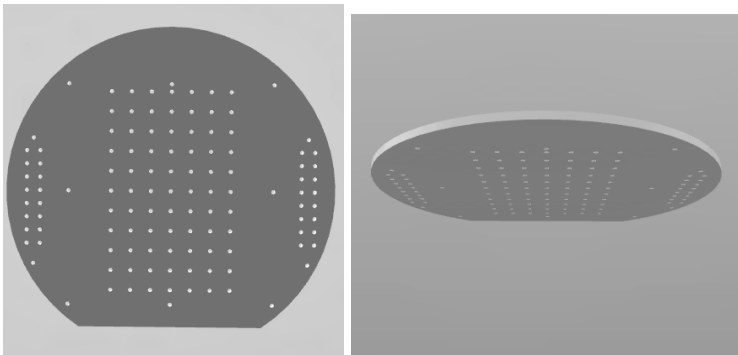
**Gambar 3.2:** Desain Robot

Dengan adanya 5 plate yang terdapat pada robot diharapkan robot tersebut dapat membawa lebih banyak perangkat. Pada robot tersebut plate 1 akan digunakan sebagai tempat baterai tambahan dan sumber daya bagi NUC dan Kinect. Plate kedua akan digunakan sebagai tempat regulator dan port daya. Plate ketiga akan digunakan sebagai tempat NUC dan USB Hub. Plate kelima akan digunakan sebagai tempat peletakan kinect. Tata letak perangkat pada robot ditunjukkan pada gambar 3.3 . Pada beberapa

plate terdapat lubang-lubang yang berfungsi sebagai tempat melekatkan perangkat pada robot dan tempat peletakan rangka besi penyambung.



**Gambar 3.3:** Tata letak perangkat pada robot



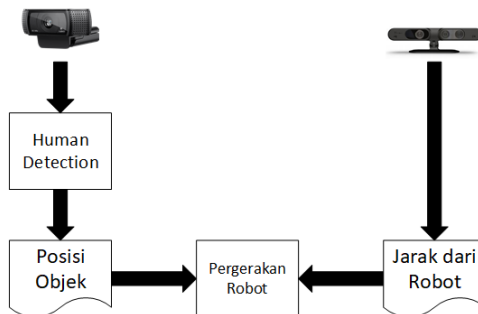
**Gambar 3.4:** Desain Plate pada Robot

### 3.3 Desain Metode Object Following

Untuk dapat melakukan proses penjajakan objek berupa manusia, maka robot harus mampu membedakan objek berupa manusia dan bukan manusia. Pada pengujian kali ini digunakan 2 metode untuk mendeteksi manusia yaitu *histogram of oriented gradient* dan *skeletal tracking*.

#### 3.3.1 *Object Following* menggunakan *Histogram of Oriented Graient*(HOG)

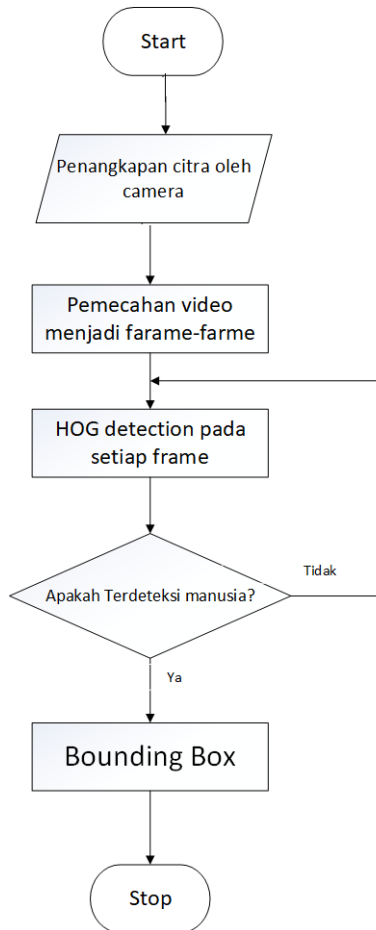
Pada sistem ini digunakan metode *histogram of Oriented Gradient* sebagai metode deteksi manusia. Dalam sistem ini digunakan 2 buah sensor yaitu logitech c920 sebagai sensor RGB dan *asus xtion pro* sebagai sensor jarak. Citra RGB hasil tangkapan kamera, diproses oleh metode HOG agar dapat membedakan antara manusia dan bukan manusia. Objek yang dianggap sebagai manusia kemudian dia cari koordinat titik tengah dari objek tersebut. Koordinat titik tengah objek akan dibandingkan dengan nilai tengah dari frame agar dapat menentukan posisi obyek tersebut dari robot apakah berada disisi kanan, tepat didepan robot, atau berada disisi kiri robot. Selain itu digunakan sensor *depth* agar dapat mengetahui jarak robot terhadap objek yang berada di depannya. Nilai jarak yang didapat akan digunakan sebagai parameter untuk pergerakan robot. Blok diagram penelitian sistem tersebut ditunjukkan pada gambar 3.5.



**Gambar 3.5:** Blok Diagram Penelitian menggunakan HOG



Pada proses deteksi manusia menggunakan HOG, pertama citra RGB dari kamera c920 dipecah kedalam frame-frame. Metode HOG diterapkan terhadap setiap frame sehingga nanti objek yang di anggap sebagai manusia akan ditandai dengan sebuah kotak pada objek tersebut.



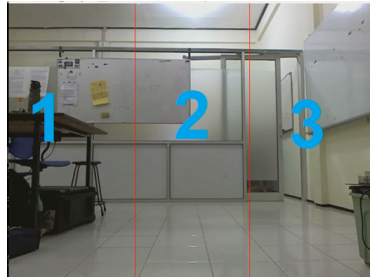
**Gambar 3.6:** Alur Deteksi Manusia

Setelah objek terdeteksi kemudian dilakukan proses untuk menentukan posisi objek tersebut dalam frame. Penentuan posisi didasarkan pada nilai titik tengah objek dengan selisih dari nilai tengah sebuah frame dengan ukuran 640 x 480 pixel berdasarkan persamaan (3.6). Pada sistem ini frame dibagi menjadi 3 area. Area pertama berada pada titik 0 sampai 220 pixel sumbu x. Area kedua berada pada titik 221 sampai 420 pixel pada sumbu x. Sedangkan area ketiga berada pada titik 421 sampai 640 pixel pada sumbu x. Objek yang telah terdeteksi sebagai manusia selanjutnya akan dicari nilai tengah dari objek tersebut dan kemudian nilai titik tengah tersebut akan dilakukan perhitungan terhadap sumbu x agar dapat diketahui posisi objek berada pada area posisi seberapa. Setelah diketahui berada pada posisi seberapa objek tersebut maka data posisi objek akan disimpan kedalam sebuah file agar dapat digunakan dalam proses pergerakan robot.

$$Posisi = \left\{ \begin{array}{ll} 1, & 0 \leq x \leq 220 \\ 2, & 221 \leq x \leq 420 \\ 3, & 421 \leq x \leq 640 \end{array} \right\} \quad (3.1)$$



**Gambar 3.7:** Proses Penentuan Posisi Objek



**Gambar 3.8:** Pembagian Daerah Posisi Objek

Sedangkan *asus xtion pro* digunakan untuk mengetahui jarak relative antara objek dengan robot. Nilai yang dapat dari sensor depth disimpan pada sebuah file kemudian digunakan untuk parameter pergerakan maju dan mundur. Dalam melakukan pergerakan robot menggunakan 2 file tersebut untuk menentukan pergerakan. Pergerakan yang dapat dilakukan robot maju, mundur, berputar kiri, berputar kanan, serta kombinasi dari keempat gerakan tersebut.

---

**Algorithm 1:** Pergerakan Robot

---

```

1: function INPUT POSISI : INT, JARAK : INT
2: if Posisi = 1 then
|   Putar kiri sebesar 0.2 rad/s
|   end
|   else
|   if Posisi = 3 then
|   |   Putar kanan sebesar 0.2 rad/s
|   |   end
|   end
3: if Posisi = 1 then
|   Putar kiri sebesar 0.2 rad/s
|   end
|   else
|   if Posisi = 3 then
|   |   Putar kanan sebesar 0.2 rad/s
|   |   end
|   end
4: end function

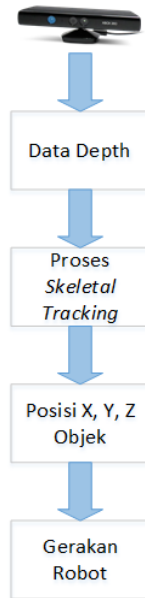
```

---

### 3.3.2 Object Following menggunakan *Skeletal Tracking*

Pada sistem deteksi manusia dengan metode *skeletal tracking*, sensor yang digunakan adalah sensor Kinect. Citra *depth* dari sensor di proses menggunakan *library* NiTE sehingga dapat membedakan objek manusia dan objek selain manusia. Setelah terdeteksi sebagai objek manusia kemudian mengambil nilai posisi objek pada sumbu-x dan sumbu-z. Nilai-nilai pada tiap sumbu tersebut digunakan sebagai parameter pergerakan robot. Blok diagram sistem ditunjukkan

oleh gambar 3.9 .



**Gambar 3.9:** Blok Diagram Penelitian menggunakan *skeletal tracking*

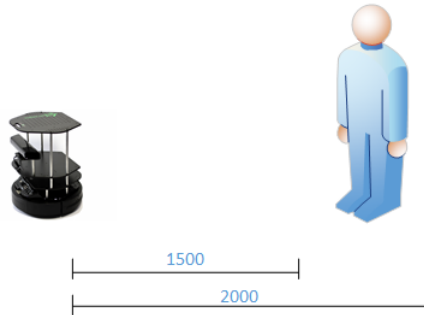
Proses pemisahan objek manusia dan bukan melalui beberapa tahapan. Pertama data yang ditangkap oleh sensor dianggap sebagai latarbelakang. Ketika terdapat perubahan posisi terhadap suatu objek atau daerah tertentu maka daerah tersebut dianggap sebagai objek. Kemudian objek tersebut akan diproses untuk menentukan apakah objek tersebut objek manusia atau bukan. Jika benar objek tersebut manusia maka akan digambar bentuk tulang pada objek tersebut sekaligus menentukan posisi sendi dan nilai pada tiap sendi pada sumbu-x, sumbu-y dan sumbu-Z.



**Gambar 3.10:** Proses deteksi manusia pada *skeletal tracking*

Nilai yang diambil untuk menentukan gerakan robot adalah bagian torso. Hal ini dikarenakan bagian torso berada ada bagian tengah objek. Nilai sumbu-x digunakan sebagai penentu arah gerakan memutar robot, sedangkan nilai pada sumbu-z digunakan untuk menentukan pergerakan maju dan mundur robot.

$$Arahputar = \begin{cases} Kiri, & x \leq -220 \\ Tetap, & -220 \leq x \leq 220 \\ Kanan, & x \geq 220 \end{cases} \quad (3.2)$$



**Gambar 3.11:** Jarak Robot dengan objek

*Halaman ini sengaja dikosongkan*

## BAB 4

### Implementasi

Pada bab ini dijelaskan mengenai implementasi atau hasil dari setiap desain yang telah dibuat pada desain sistem.

#### 4.1 Implementasi Robot

Pada bagian ini akan dijelaskan hasil dari pembuatan robot berdasarkan desain yang telah dibuat, termasuk susunan perangkat pada robot.



**Gambar 4.1:** Bentuk Robot

Gambar 4.1 menunjukkan hasil robot yang telah selesai dibuat dan dirakit. Robot yang memiliki ketinggian 850 mm dan diameter 354 mm ini mempunyai 5 rak atau tempat yang dapat digunakan untuk meletakkan perangkat pada robot. Pada tiap rak terdapat

lubang-lubang yang berfungsi sebagai tempat peletakan mur dan tempat pengikat perangkat pada robot agar tidak jatuh (gambar 4.3 ). Sedangkan gambar 4.2 menunjukan robot yang telah dipasangi perangkat.



**Gambar 4.2:** Tata letak perangkat pada robot



**Gambar 4.3:** Desain Plate pada Robot

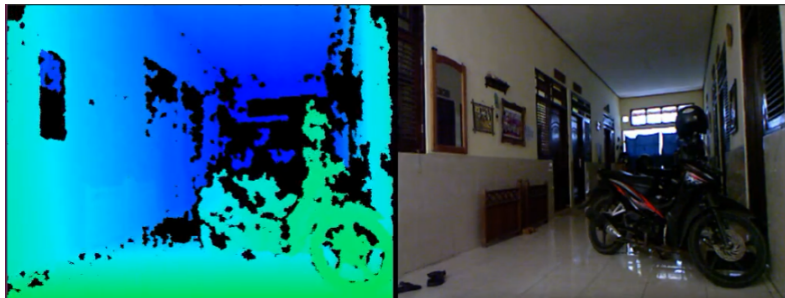


## 4.2 Implementasi Metode Object Following

Berdasarkan *block* diagram sistem yang telah dibuat diatas dilakukan proses ujicoba untuk mengetahui hasil dari desain sistem tersebut.

### 4.2.1 Implemetasi Deteksi dengan metode HOG

Pada pada sistem ini diguakan dua buah sensor yaitu *logitech c920* dan sensor depth asus *xtion pro*. Karena pada penelitian ini digunakan 2 buah sensor maka pada sistem ini data yang diolah terdapat dua macam yaitu citra RGB dan depth. Pada depth perbedaan jarak pada tiap objek direpresentasikan dalam perbedaan warna.



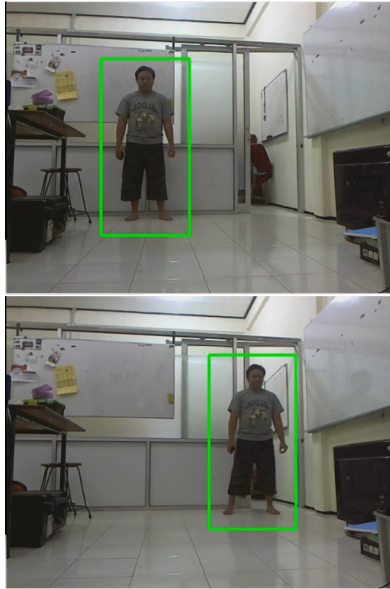
Citra Depth

Citra RGB

**Gambar 4.4:** Citra RGB dan citra *depth*

Citra RGB yang didapat dari sensor kemudian diproses dengan metode HOG. Sehingga pada citra tersebut, dapat membedakan objek yang termasuk manusia dan bukan manusia. Objek yang dianggap sebagai manusia akan diberi anda oleh sebuah persegi berwarna hijau (Gambar 4.5).

Setelah objek terdeteksi, dilakukan proses untuk menentukan posisi objek pada tangkapan layar. Pada proses ini posisi objek yang awalnya berupa posisi pixel kemudian di proses dengan persamaan 3.1 sehingga menghasilkan posisi 1, 2, dan 3. Hasil dari penentuan posisi dapat dilihat pada gambar 4.6 dan 4.7 .



Gambar 4.5: Hasil deteksi menggunakan HOG



Gambar 4.6: Hasil objek pada posisi 1



**Gambar 4.7:** Hasil objek pada posisi 2

Sensor depth pada proses ini digunakan untuk mengetahui jarak antara robot dan benda yang ada di depannya. Hasil jarak ini akan digunakan sebagai penentu pergerakan maju, diam, dan mundur pada robot.



**Gambar 4.8:** Hasil ketika objek jauh



**Gambar 4.9:** Hasil ketika objek berada di posisi ideal



**Gambar 4.10:** Hasil ketika objek terlalu dekat

## 4.2.2 Implementasi pada metode *Skeletal Tracking*

1. Hasil dari sensor depth

Gambar 4.11 menunjukkan hasil tangkapan yang diperoleh dari sensor depth. Perbedaan warna keabu-abuan pada citra ini

menunjukkan jarak objek dengan sensor. jika jarak objek dekat dengan sensor maka akan berwarna keputih-putihan dan jika objek jauh dari sensor maka akan berwarna lebih gelap mendekati hitam.



**Gambar 4.11:** Hasil tangkapan sensor depth

## 2. Proses Deteksi

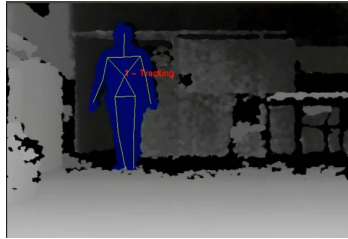
Ketika ada objek yang bergerak atau perubahann posisi pada objek yang tertangkap oleh sensor maka pada bagian tersebut akan dilakukan proses kalibrasi.



**Gambar 4.12:** Proses kalibrasi pada objek tersebut

## 3. Proses tracking objek

Setelah berhasil di kalibrasi dan bahwa objek tersebut berupa manusia maka objek tersebut akan dilakukan proses *tracking* dan menggambar struktur tulang dan sendi pada objek tersebut.



**Gambar 4.13:** Proses tracking objek

Ketika terjadi proses *tracking* kita dapat mengetahui posisi objek dan jarak objek tersebut dari sensor. Hal ini dapat dilakukan dengan cara menggunakan nilai yang didapat dari tiap-tiap sendi yang disediakan oleh *library*.



**Gambar 4.14:** Hasil Skeletal tracking dengan nilai beberapa joint pada sumbu x,y,z

## BAB 5

### PENGUJIAN DAN ANALISIS

Pada bab ini dijelaskan mengenai hasil pengujian Implementasi *Metode Tracking Object* pada *Robot Service* dan analisis tingkat keberhasilannya terkait pengujian-pengujian yang dilakukan.

#### 5.1 Mekanisme Pengujian

Pengujian Implementasi *Metode Tracking Object* pada *Robot Service*. Pada penelitian ini dilakukan beberapa pengujian, diantaranya yaitu :

##### 5.1.1 Pengujian Robot

Pengujian robot ini ditujukan agar dapat mengetahui kemampuan robot dan batas maksimal yang dapat dilakukan robot.

1. Pengujian kecepatan robot

Pengujian ini dilakukan untuk mengetahui batas kecepatan aman robot agar robot tidak jatuh. Tahapan pengujian ini dengan cara melakukan pergerakan robot pada lintasan sepanjang 3 meter dengan kecepatan konstan.

2. Pengujian charger otomatis

Pengujian ini dilakukan untuk mengetahui kemampuan robot dalam melakukan charging otomatis saat daya baterai pada robot tinggal sedikit.

##### 5.1.2 Pengujian Metode *Histogram of Oriented Gradient* (HOG)

Pengujian metode HOG sebagai deteksi manusia object following terbagi kedalam beberapa pengujian. Pengujian dalam tahap ini terbagi kedalam beberapa pengujian dibawah ini.

1. Pengujian pada resolusi yang berbeda

Proses pendeteksian dilakukan pada resolusi 640x480, 480x320, dan 320x240 dengan 30 fps. Tujuan pengujian ini mengetahui dampak ukuran resolusi yang berbeda pada hasil deteksi dan waktu proses.

2. Pengujian Metode HOG pada jarak tertentu

Metode HOG keberhasilan deteksi pada beberapa jarak. Jarak yang digunakan adalah 1 meter, 2 meter, 3 meter, 4 meter, dan 5 meter.

### **5.1.3 Pengujian Metode *Skeletal Tracking***

Pengujian pada metode *skeletal tracking* berupa pengujian jangkauan yang dapat ditangkap oleh sensor dan dapat diproses. Tujuan pengujian untuk mengetahui kisaran jarak yang dapat dijangkau oleh metode ini. Jarak yang digunakan pada pengujian ini adalah 1 meter, 2 meter, 3 meter, 4 meter dan 5 meter.

### **5.1.4 Pengujian Metode HOG dan *Skeletal Tracking* pada berbagai Pose Tubuh**

Pengujian pada metode HOG dan *Skeletal Tracking* pada pose tubuh yang berbeda untuk mengetahui pose-pose yang tidak dapat di deteksi oleh metode HOG. Pose yang digunakan pada pengujian sebanyak 9 posisi, yang terdiri dari pose berdiri hadap kamera, berdiri membeakangi kamera, berdiri menghadap samping, duduk menghadap kamera, duduk membelakangi kamera, duduk menghadap samping, membungkuk hadap kamera, membungkuk hadap samping, dan membungkuk membelakangi kamera.

### **5.1.5 Pengujian Kemampuan Robot Melakukan Penjajakan**

Pengujian pada tahap ini bertujuan untuk mengetahui kemampuan robot dalam melakukan penjajakan. Terdapat 3 macam pengujian pada bagian ini yaitu, pengujian kemampuan robot dalam melakukan penjajakan terhadap objek, pengujian kemampuan robot dalam menyesuaikan kecepatan, dan kemampuan robot dalam melakukan penjajakan ketika terdapat 2 orang yang terdeteksi oleh robot

### **5.1.6 Pengujian Penjajakan ketika terdapat bayangan dari objek**

Pengujian ini bertujuan untuk mengetahui hasil penjajakan jika objek memiliki bayangan hasil dari pencahayaan. Dalam pengujian ini akan dilihat apakah bayangan juga ikut terdeteksi.



### 5.1.7 Pengujian Penjajakan ketika Objek Berjalan Menggunakan Alat Bantu

Pengujian ini dilakukan untuk mengetahui proses penjajakan ketika objek berjalan menggunakan alat bantu. Dalam pengujian ini apakah robot dapat mendeteksi dengan benar dan melakukan penjajakan.

### 5.1.8 Pengujian Penjajakan Diluar Ruangan (*Outdoor*)

Tujuan dari pengujian ini untuk mengetahui jika proses penjajakan dilakukan diluar ruangan. Selain itu pengujian ini juga digunakan untuk menganalisa faktor-faktor yang mengganggu proses penjajakan.

## 5.2 Hasil dan Pembahasan

Berikut merupakan hasil dan pembahasan terhadap percobaan yang telah dilakukan.

### 5.2.1 Pengujian Robot

#### 1. Pengujian kecepatan robot

Pada pengujian ini robot bergerak dengan kecepatan tertentu pada jarak lintasan 3 meter. Dalam pengujian ini digunakan untuk mengetahui kecepatan maksimum robot agar dapat bergerak dengan aman.

**Tabel 5.1:** Hasil Pengukuran pergerakan robot

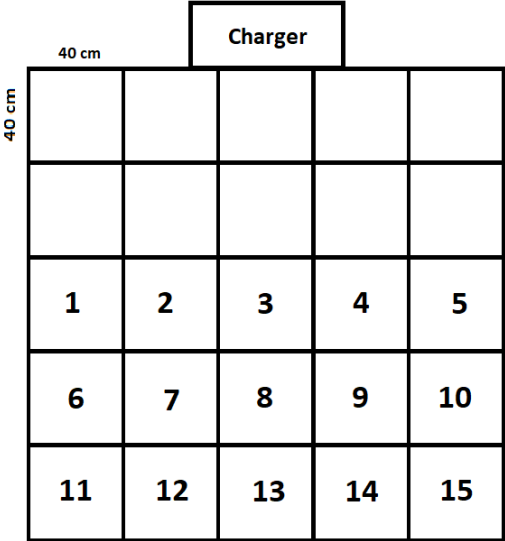
No	Percobaan	Kecepatan	Keterangan
1	Percobaan 1	0,2	Aman
2	Percobaan 2	0,4	Aman
3	Percobaan 3	0,6	Aman
4	Percobaan 4	0,8	Aman
5	Percobaan 5	1	Aman
6	Percobaan 6	1,2	Sedikit Berguncang
7	Percobaan 7	1,4	Berguncang
8	Percobaan 8	1,6	Hampir Roboh
9	Percobaan 9	1,8	Robot tidak bergerak

Berdasarkan data pada tabel 5.1 dapat dilihat bahwa batas

aman kecepatan robot bergerak dibawah 1.2 m/s. Ketika robot bergerak diatas 1.2 m/s terjadi guncangan pada bagian atas robot. Sedangkan ketika dicoba pada kecepatan 1.8 m/s robot tidak bergerak. Hal ini diakibatkan pada nilai tersebut, sistem merasa nilai yang diberikan melebihi nilai maksimal yang dapat digunakan sehingga robot tidak bergerak.

2. Pengujian *charger* otomatis

Pengujian ini dilakukan untuk mengetahui kemampuan robot dalam melakukan *charging* otomatis saat daya baterai pada robot tinggal sedikit. Robot akan diletakkan pada beberapa posisi untuk mengetahui jarak dan kemampuan robot dalam melakukan pengisian daya otomatis. Tata letak posisi pengujian akan dapat dilihat pada gambar 5.1



**Gambar 5.1:** Peta percobaan pengisian daya otomatis

Hasil percobaan pengisian daya otomatis terlampir pada tabel 5.2 . Pada tabel tersebut dari 15 posisi yang di uji, terdapat 2 posisi yang gagal yaitu pada posisi 1 dan posisi 5. Dalam posisi tersebut robot tidak berhasil mencapai tempat untuk mengisi daya. Hal ini disebabkan oleh jangkauan sinar inframerah pada tempat pengisian daya yang memiliki sudut kecil. Sehingga ketika robot berada diluar jangkauan sinar *inframera* maka robot kebingungan mencari keberadaan pengisian daya. Dapat disimpulkan pula bahwa pada posisi 1 dan 5 merupakan daerah titik buta bagi robot untuk mendeteksi keberadaan pengisian daya otomatis.

**Tabel 5.2:** Hasil Pengukuran pergerakan robot

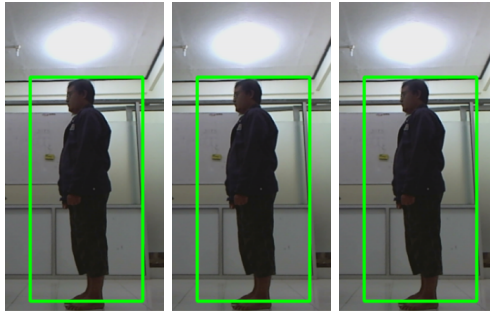
No	Posisi	Keterangan
1	Posisi 1	gagal
2	Posisi 2	berhasil
3	Posisi 3	berhasil
4	Posisi 4	berhasil
5	Posisi 5	gagal
6	Posisi 6	berhasil
7	Posisi 7	berhasil
8	Posisi 8	berhasil
9	Posisi 9	berhasil
10	Posisi 10	berhasil
11	Posisi 11	berhasil
12	Posisi 12	berhasil
13	Posisi 13	berhasil
14	Posisi 14	berhasil
15	Posisi 15	berhasil

### 5.2.2 Pengujian Metode *Histogram of Oriented Gradient* (HOG)

#### 1. Resolusi yang berbeda

Berdasarkan percobaan resolusi yang berbeda tidak berpengaruh pada proses deteksi. Pada resolusi 640x480, 480x320, dan 320,240 semua objek terdeteksi dengan baik (gambar 5.2).

Namun perbedaan resolusi berdampak pada waktu yang dibutuhkan dalam memproses tiap frame. Berdasarkan data pada tabel 4.1, semakin besar resolusi waktu yang dibutuhkan untuk memproses 1 buah frame semakin besar. Besar nilai waktu untuk memproses satu buah frame berdampak pada pergerakan robot. Semakin besar waktu yang digunakan untuk memproses tiap frame membuat pergerakan robot terlihat lambat dalam merespon gerakan objek.



**Gambar 5.2:** gambar a proses deteksi pada resolusi 640x480, gambar b proses deteksi pada 480x320, gambar c proses deteksi pada resolusi 320x240

**Tabel 5.3:** Waktu dalam memproses tiap Frame



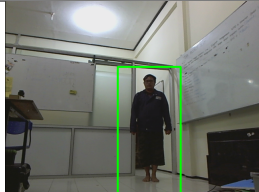
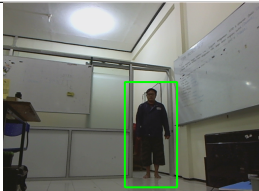
No	Resolusi	waktu rata-rata dalam 100 frame
1	640 x 480	0,7681 detik
2	480 x 320	0,2141 detik
3	320 x 240	0,1703 detik

Berdasarkan data pada tabel 4.1 waktu yang diperlukan memproses sebuah frame dalam resolusi 640x480 membutuhkan waktu yang lebih besar dari resolusi 480x320 dan 320x240. Hal ini disebabkan semakin besar resolusi semakin luas wilayah yang akan diproses oleh HOG detector, sedangkan size *block* HOG mempunyai ukuran yang tetap. Sehingga pada resolusi yang semakin besar proses untuk mengolah sebuah frame


dibutuhkan waktu yang lebih lama.

2. Pengujian deteksi HOG pada beberapa jarak  
Hal ini dilakukan untuk mengetahui jarak minimum dan maksimum agar objek dapat terdeteksi pada metode HOG.

**Tabel 5.4:** Hasil pendeteksian HOG pada beberapa macam jarak (bagian 1)

No	Jarak	Hasil	Keterangan
1	1 meter		tidak terdeteksi
2	2 meter		terdeteksi
3	3 meter		terdeteksi
4	4 meter		terdeteksi

**Tabel 5.5:** Hasil pendeteksian HOG pada beberapa macam jarak (bagian 2)

No	Jarak	Hasil	Keterangan
5	5 meter		terdeteksi


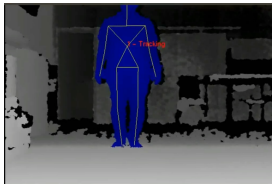
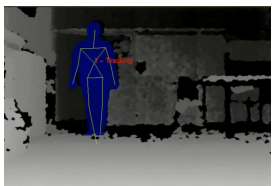

Berdasarkan data pada tabel 4.4, pada jarak 2 meter, 3 meter, 4 meter dan 5 meter deteksi HOG dapat berjalan dengan baik, sedangkan pada jarak 1 meter deteksi HOG tidak dapat berjalan. Hal ini disebabkan pada jarak 2 meter, 3 meter, 4 meter, dan 5 meter bentuk tubuh terlihat sempurna tanpa ada bagian yang terpotong sehingga ketika data HOG descriptor dibandingkan dengan data set yang dimiliki oleh HOG perbedaan tidak terlalu jauh dan hasilnya dapat terdeteksi dengan benar. Pada pengujian ini jarak terjauh yang digunakan adalah 5 meter, pada jarak tersebut HOG masih dapat mendeteksi dengan baik sehingga dapat dikatakan pada jarak 5 meter proses deteksi menggunakan HOG hasil terdeteksi dengan baik dan jarak terdekat agar dapat terdeteksi adalah 1,5 meter.

### 5.2.3 Pengujian Metode *Skeletal Tracking*


1. Pengujian deteksi *skeletal tracking* pada jarak tertentu  
Agar dapat melakukan proses penajakan terus-menerus robot tidak boleh kehilangan objek atau objek tidak dapat dijangkau oleh sensor. Pengujian ini dilakukan agar dapat mengetahui berapa jarak terjauh dan jarak terdekat agar dapat terdeteksi. Jarak yang digunakan pada pengujian kali ini adalah 1 meter, 2 meter, 3 meter, 4 meter, dan 5 meter.  
Pada tabel 4.5 terlihat bahwa pada jarak 5 meter sistem sudah tidak dapat mengenali objek dengan baik. Kemampuan kinect dalam mengenali objek dengan menggunakan *skeletal tracking* terbatas pada jarak terjauh 5 meter pada jarak ter-

jauh. sedangkan pada jarak 1 meter kinect juga tidak dapat melakukan proses deteksi karena objek terlalu dekat terhadap sensor. Sehingga dapat dikatakan jarak 1 meter proses deteksi tidak berjalan.

**Tabel 5.6:** Hasil pendeteksian *skeletal tracking* terhadap objek pada jarak tertentu (bagian 1)

No	Jarak	Hasil	keterangan
1	1 meter		tidak terdeteksi
2	2 meter		terdeteksi
3	3 meter		terdeteksi
4	4 meter		terdeteksi

**Tabel 5.7:** Hasil pendeteksian *skeletal tracking* terhadap objek pada jarak tertentu (bagian 2)

No	Jarak	Hasil	keterangan
5	5 meter		tidak terdeteksi

### 5.2.4 Metode HOG dan *skeletal tracking* pada Berbagai Posisi Tubuh

Metode HOG yang digunakan sebagai metode untuk mendeteksi manusia dianalisa tingkat keberhasilan pendeteksian pada objek manusia. Posisi-posisi yang digunakan untuk pengujian yaitu, berdiri tegak hadap kamera, berdiri tegak hadap samping, berdiri tegak membelakangi kamera, berdiri bungkuk menghadap kamera, berdiri bungkuk menghadap samping, berdiri bungkuk membelakangi kamera, duduk menghadap kamera, duduk menghadap samping, duduk membelakangi kamera, membungkuk menghadap kamera, membungkuk menghadap samping, dan membungkuk membelakangi kamera. (gambar 5.3).




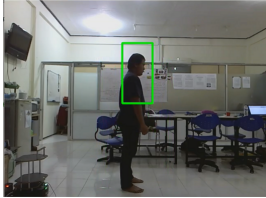






**Gambar 5.3:** Macam-macam Posisi yang akan diuji

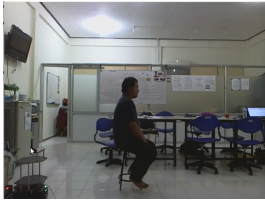
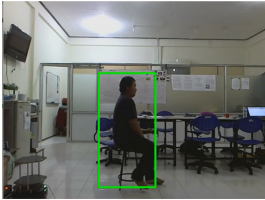

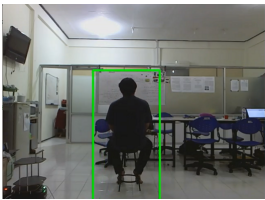




Pada tiap pose tersebut dilakukan proses deteksi manusia dengan metode HOG dan *skeletal tracking*.





**Tabel 5.8:** Hasil Deteksi dengan menggunakan metode HOG (bagian 1)

No	Posisi	Hasil	Keterangan
1			Terdeteksi Sempurna
2			Terdeteksi Sebagian
3			Terdeteksi Sempurna
4			Salah Deteksi

**Tabel 5.9:** Hasil Deteksi dengan menggunakan metode HOG (bagian 2)

No	Posisi	Hasil	Keterangan
5			Terdeteksi Sempurna
6			Terdeteksi Sempurna
7			Terdeteksi Sempurna
8			Terdeteksi Sebagian

**Tabel 5.10:** Hasil Deteksi dengan menggunakan metode HOG (bagian 3)

No	Posisi	Hasil	Keterangan
9			Terdeteksi Sempurna


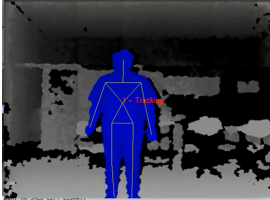

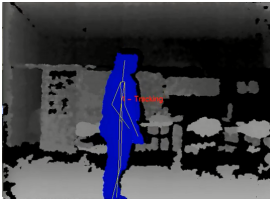

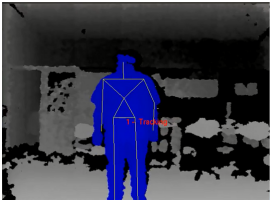

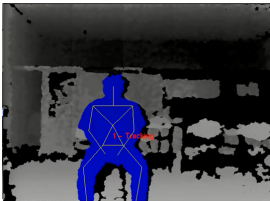
Berdasarkan pada tabel 4.2 dan 4.3 hasil deteksi pada 9 posisi yang telah di tetapkan tidak semua posisi dapat terdeteksi secara sempurna, terdapat 3 posisi yang tidak dapat terdeteksi secara sempurna. Pada posisi nomer 2 dan 8 hasil deteksi hanya terdeteksi sebagian dari objek. Sedangkan pada posisi nomer 4 terjadi kesalahan deteksi.

Hal ini disebabkan karena hasil proses dari metode HOG yang berupa HOG *descriptor* pada objek dan latarbelakang objek memiliki kemiripan sehingga SVM menganggap bagian tersebut adalah objek yang dianggap sebagai orang. Selain itu kesalahan juga dapat disebabkan karena data training yang digunakan kurang bervariasi sehingga pada pose-pose tertentu tidak dapat terdeteksi atau salah dalam mendeteksi. Pada HOG benda-benda di sekitar objek juga dapat mempengaruhi keberhasilan proses pendeteksian.


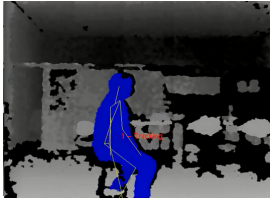

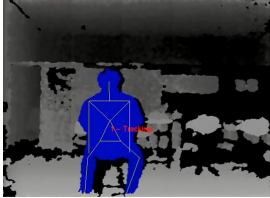

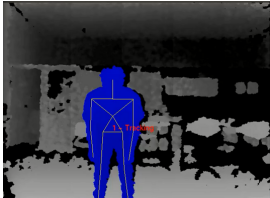

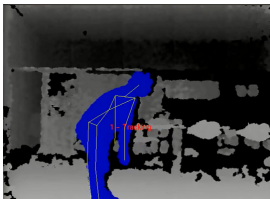


**Gambar 5.4:** Hasil Deteksi HOG yang kurang tepat


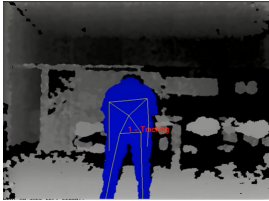
**Tabel 5.11:** Hasil Deteksi dengan menggunakan metode *Skeletal Tracking* (bagian 1)

No	Posisi	Hasil	Keterangan
1			Terdeteksi Sempurna
2			Terdeteksi Sempurna
3			Terdeteksi Sempurna
4			Terdeteksi Sempurna

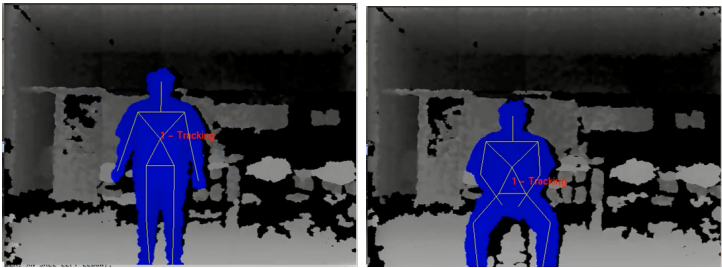
**Tabel 5.12:** Hasil Deteksi dengan menggunakan metode *Skeletal Tracking* (bagian 2)

No	Posisi	Hasil	Keterangan
5			Terdeteksi Sempurna
6			Terdeteksi Sempurna
7			Terdeteksi Sempurna
8			Terdeteksi Sempurna

**Tabel 5.13:** Hasil Deteksi dengan menggunakan metode *Skeletal Tracking* (bagian 3)

No	Posisi	Hasil	Keterangan
9			Terdeteksi Sempurna

Pada pendeteksian dengan menggunakan *skeletal tracking* dari 9 pose yang ada kesembilan pose tersebut dapat dideteksi dengan tepat. Keakurasian pada proses pendeteksian ini dikarenakan pada *skeletal tracking* terlebih dahulu dipisahkan antara *scene* atau latarbelakang dan objek. Objek yang didapat kemudian objek tersebut dikalibrasi untuk menentukan apakah objek tersebut merupakan manusia atau bukan manusia. Jika proses kalibrasi berhasil dan objek tersebut merupakan manusia maka objek tersebut akan di *tracking* kemanapun selama berada dalam wilayah frame tersebut. Sehingga meskipun objek tersebut bergerak dengan posisi yang berbeda-beda selama didalam area jangkauan tangkapan kamera objek akan selalu terdeteksi dengan benar.



**Gambar 5.5:** Hasil deteksi menggunakan *skeletal tracking*

### 5.2.5 Pengujian Penjajakan oleh Robot

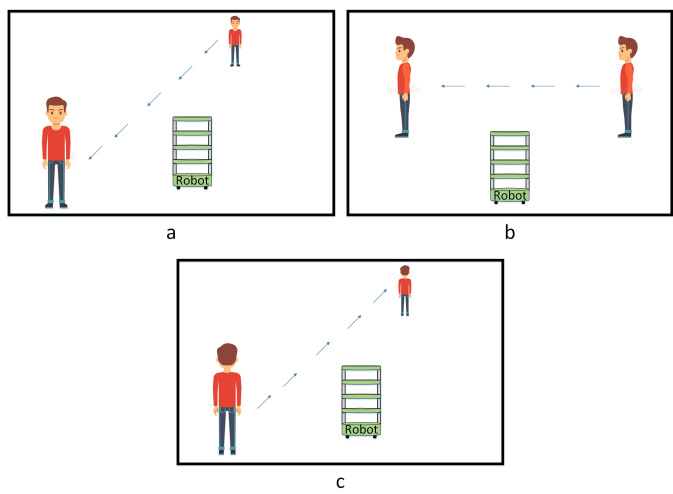
Berdasarkan hasil pengujian sebelumnya didapat bahwa hasil deteksi orang menggunakan *skeletal tracking* lebih baik dari pada deteksi orang menggunakan metode HOG, maka pada proses *tracking robot*, deteksi objek yang digunakan adalah metode *skeletal tracking*. Pengujian ini bertujuan untuk menguji kemampuan robot dalam melakukan penjajakan, pengujian yang dilakukan meliputi kemampuan robot dalam melakukan penjajakan, kemampuan robot dalam menyesuaikan kecepatan robot dengan berdasarkan besar jarak dengan objek, dan kemampuan robot dalam melakukan penjajakan ketika ada 2 orang yang terdeteksi.

#### 1. Kemampuan robot dalam melakukan penjajakan

Pengujian ini bertujuan untuk mengetahui bagaimana hasil robot dalam melakukan penjajakan manusia. Dalam pengujian ini dilakukan percobaan pada robot dalam mengikuti pergerakan orang. Pengujian ini dilakukan di dalam ruangan berbentuk persegi yang bebas (tanpa ada halangan). Pada percobaan pertama, dilakukan percobaan untuk mendeteksi objek dari posisi depan, samping, dan belakang, pada ketiga posisi tersebut apakah robot dapat melakukan penjajakan. Proses deteksi dari depan, orang atau objek posisi menghadap robot kemudian berjalan melewati depan robot. Pada proses deteksi dari samping, objek akan bergerak dari sisi kanan robot menuju sisi kiri robot. Dalam proses tersebut objek terlihat oleh robot pada bagian samping saja. Proses deteksi dari belakang hampir sama dengan skema deteksi dari depan, yang berbeda hanya arah gerak dan posisi badan objek. Desain percobaan dapat dilihat pada Gambar 5.6.

Hasil dari percobaan ini menunjukkan bahwa proses deteksi pada posisi depan, samping, dan belakang berhasil dilakukan oleh robot. Robot berhasil melakukan penjajakan pada objek, meskipun objek datang dari arah yang berbeda-beda. Namun pada deteksi dengan posisi objek menghadap ke robot memerlukan waktu yang lebih lama dari pada posisi dari samping

dan posisi objek membelakangi robot. Tabel 5.12 menunjukkan waktu yang diperlukan untuk proses deteksi objek. Selama 5 kali percobaan, waktu rata-rata yang diperlukan pada ketika mendeteksi dari sisi depan lebih besar dari pada dari sisi samping dan belakang. Hal ini disebabkan objek bergerak dari tempat yang jauh menuju ke arah robot. Semakin jauh objek proses untuk deteksi membutuhkan waktu yang lebih lama karena pada saat jauh objek terlihat kecil dan samar-samar.

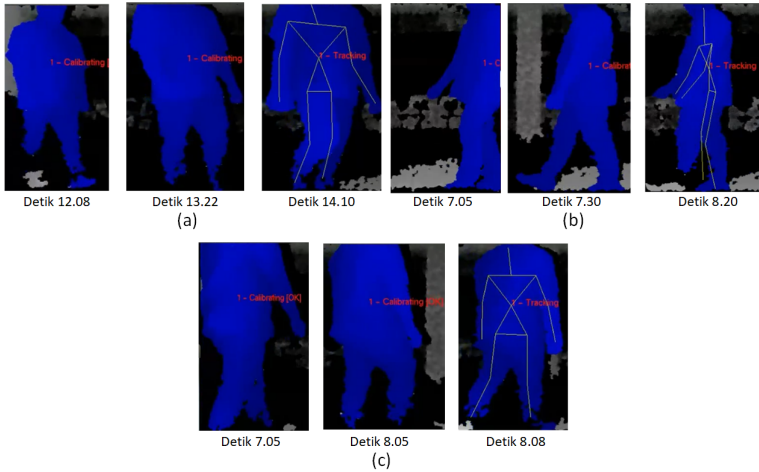


**Gambar 5.6:** Skema pengujian robot dalam mendeteksi objek. Gambar (a) mendeteksi objek dari depan, (b) mendeteksi objek dari samping, (c) mendeteksi objek dari belakang

**Tabel 5.14:** Waktu dalam proses deteksi objek

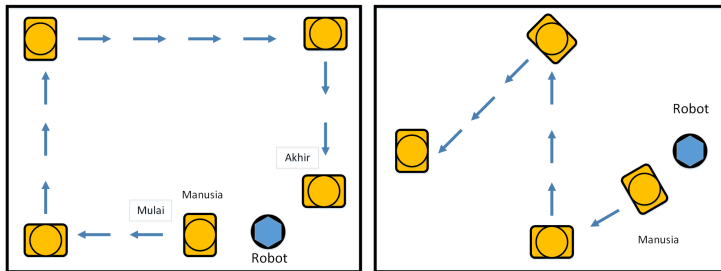
No	Percobaan	Posisi Depan	Posisi Miring	Posisi Belakang
1	Percobaan ke-1	2.02 detik	1.15 detik	1.03 detik
2	Percobaan ke-2	3 detik	2.06 detik	1.03 detik
3	Percobaan ke-3	3.17 detik	1.91 detik	1.88 detik
4	Percobaan ke-4	2,75 detik	0.97 detik	1.23 detik
5	Percobaan ke-5	2,97 detik	1.05 detik	1,19 detik
	Rata-Rata	2,782 detik	1,428 detik	1,272 detik





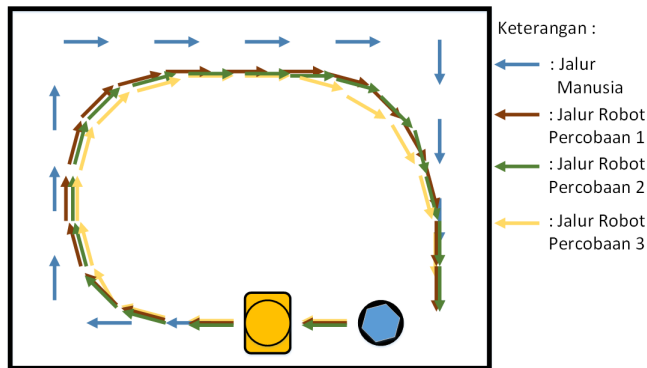
**Gambar 5.7:** Gambar (a) hasil deteksi dari depan, (b) hasil deteksi dari samping, (c) hasil deteksi dari belakang

Selanjutnya pengujian kemampuan robot dalam melakukan penjajakan. Pada pengujian ini terdapat 2 model pengujian, pertama objek akan berjalan mengelilingi ruangan yang berbentuk persegi dan yang kedua objek akan berjalan zigzag dalam ruangan tersebut. Skema pengujian kemampuan robot dalam melakukan penjajakan dapat dilihat pada Gambar 5.8 .



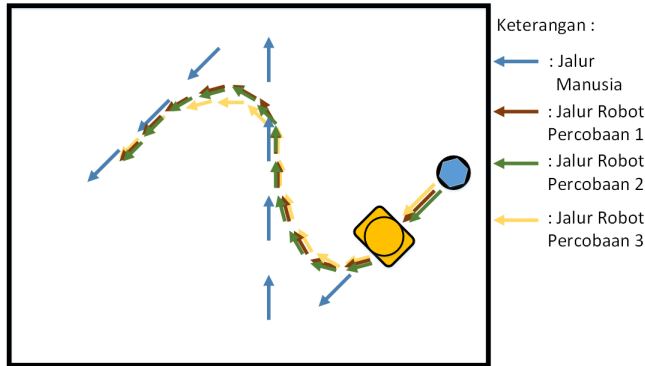
**Gambar 5.8:** Skema pengujian kemampuan penjajakan objek oleh robot *skeletal tracking*

pengujian ketika objek bergerak mengelilingi ruangan robot dapat bergerak mengikuti objek, meskipun jalur pergerakan robot lebih pendek daripada jalur yang dilewati oleh objek (Gambar 5.9). Pergerakan robot pada pengujian ini memiliki jalur yang lebih pendek yang di akibatkan oleh jarak antara objek dan robot. Hasil 3 kali pengujian robot melakukan penjajakan objek mengelilingi ruangan, menunjukkan bahwa jalur pergerakan robot hampir memiliki kemiripan. Pergerakan robot pada pengujian ini semakin jauh jarak antara objek dan robot maka pergerakan robok akan berbentuk oval.



**Gambar 5.9:** Jalur pergerakan robot selama melakukan penjajakan terhadap objek

Pada pengujian dengan jalur zigzag robot dapat melakukan penjajakan dari awal hingga akhir. Robot dapat mengikuti objek terus-menerus sesuai dengan pergerakan objek. Jalur hasil pergerakan robot dapat dilihat pada Gambar 5.10 . Pada gambar tersebut, hasil dari 3 kali percobaan robot dapat mengikuti objek hingga akhir. Seperti hasil pada percobaan mengelilingi ruangan, jalur yang dilalui oleh robot saat pengujian pergerakan zigzag mengalami perbedaan ketika objek bergerak berubah arah. Pada saat itu robot akan mengambil jalur terpendek agar tidak kehilangan objek tersebut.



**Gambar 5.10:** Jalur robot selama melakukan penjajakan dengan jalur zigzag

Selain itu dilakukan uji coba penjajakan dalam keadaan pencahayaan normal dan gelap. Pada keadaan gelap robot dapat melakukan penjajakan dengan baik seperti dalam keadaan pencahayaan normal. Hal ini disebabkan oleh sensor Depth yang digunakan dalam deteksi manusia. Sensor Depth memiliki keunggulan dalam menangkap objek dalam keadaan pencahayaan yang minim. Sehingga perubahan pencahayaan tidak berpengaruh dalam proses pengakapan citra oleh sensor Depth.

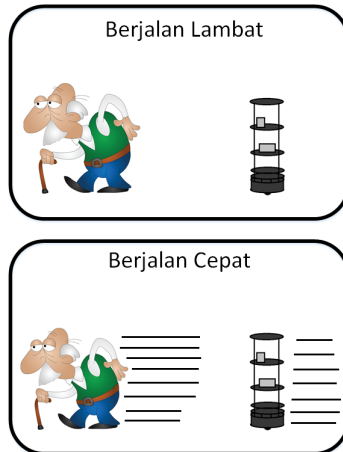


**Gambar 5.11:** Hasil penjajakan dengan pencahayaan normal



**Gambar 5.12:** Hasil penjajakan dengan pencahayaan gelap

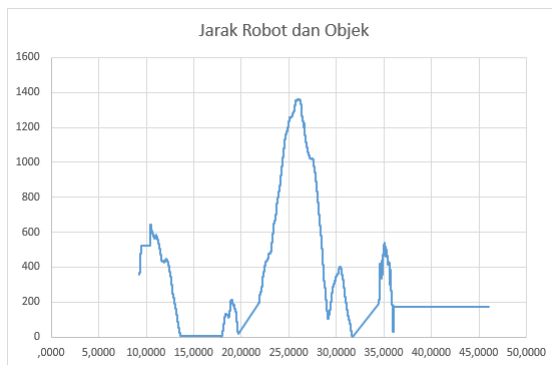
2. Kemampuan robot dalam menyesuaikan kecepatan gerak  
 Pengujian ini bertujuan agar dapat mengetahui kemampuan robot untuk menyesuaikan kecepatan gerak terhadap kecepatan gerak objek. Apabila jarak antara robot dan objek terlalu jauh dari jarak ideal maka robot akan meningkatkan kecepatan hingga jarak antara objek dan robot kembali ideal (sekitar 1.3 meter). Hal ini dilakukan agar jarak robot dan objek tidak terlalu jauh dan menyebabkan robot kehilangan objek.



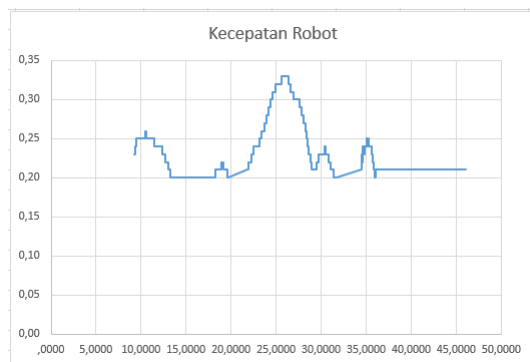
**Gambar 5.13:** Ilustrasi pengujian kecepatan robot

Gambar 5.14 dan 5.15 merupakan hasil dari pengujian ini, gambar 5.14 merupakan hasil besaran jarak terhadap waktu,

sedangkan gambar 5.15 merupakan kecepatan robot terhadap waktu. Sebenarnya robot memiliki kecepatan minimum sebesar 0.2 m/s, namun kecepatan tersebut dapat berubah menjadi lebih cepat ketika jarak robot dan objek semakin besar. Berdasarkan gambar 5.14 dan 5.15, terlihat bahwa kecepatan robot dipengaruhi oleh jarak antara robot dan objek. Ketika jarak terlalu besar maka kecepatan robot akan bertambah semakin tinggi. Hal ini bertujuan untuk menjaga jarak robot seideal mungkin sehingga robot tidak kehilangan objek tersebut.



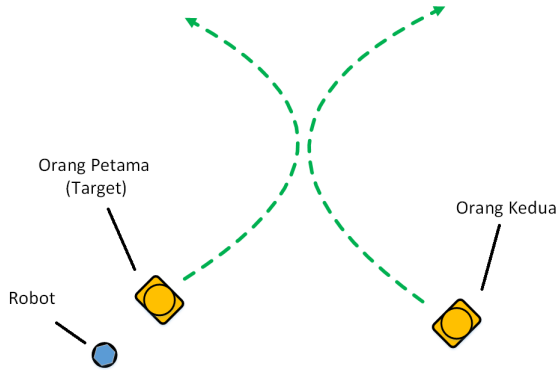
**Gambar 5.14:** Grafik jarak objek dengan robot terhadap waktu



**Gambar 5.15:** grafik kecepatan robot terhadap waktu

3. Kemampuan robot dalam melakukan penjajakan ketika ada 2 orang yang terdeteksi

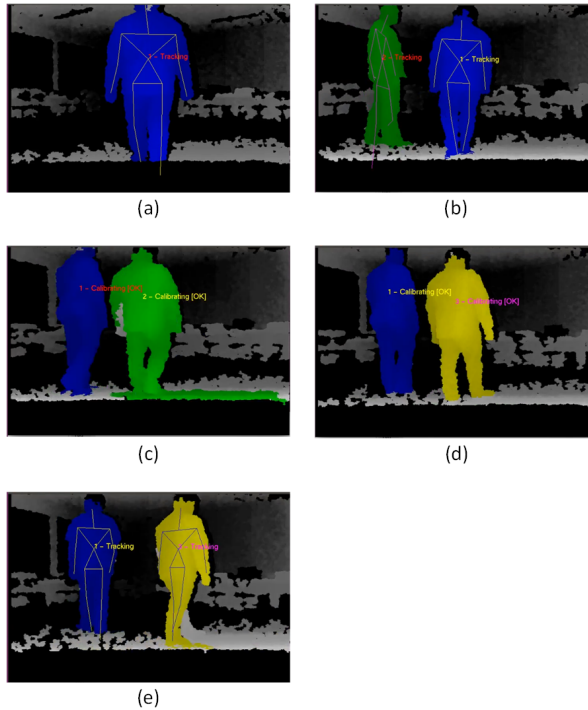
Pengujian ini dilakukan untuk mengetahui apabila terdapat 2 orang yang terdeteksi oleh robot disaat melakukan penjajakan. Skema dalam pengujian ini dapat dilihat pada Gambar 5.16 . Pada gambar tersebut mula-mula robot melakukan penjajakan terhadap objek, kemudian beberapa saat datang orang lain dan juga terdeteksi oleh robot. Kedua orang tersebut bertemu di satu titik dan setelah itu mereka menyebar. Pada saat menyebar kita akan melihat robot akan mengikuti orang yang sebelumnya dilakukan penjajakan atau mengikuti orang yang tiba-tiba muncul.



**Gambar 5.16:** Skema pengujian 2 orang

Hasil dari 5 kali percobaan menunjukan bahwa robot tidak dapat melakukan penjajakan hingga akhir. Mula-mula robot dapat melakukan penjajakan dengan normal, kemudian saat orang kedua masuk dan terdeteksi oleh robot, robot masih dapat melakukan penjajakan terhadap orang pertama. Saat kedua objek akan bertemu atau berdekatan tiba-tiba robot berhenti bergerak. Hal ini disebabkan ketika melakukan proses penjajakan beberapa kali robot mengalami proses deteksi ulang terhadap objek. Ketika terdapat objek lain yang berada di tempat yang hampir sama dengan objek target dimungkinkan terjadi perubahan pemberian identitas pada kedua objek

tersebut. Sehingga menyebabkan objek yang diikuti oleh robot berubah.



**Gambar 5.17:** Hasil deteksi menggunakan *skeletal tracking* pada saat terdapat 2 orang

Gambar 5.17 (a) menunjukkan proses tracking yang dilakukan oleh robot pada saat objek masih sendiri. Kemudian gambar 5.17 (b) menunjukkan hasil yang dilakukan oleh robot ketika terdapat objek lain yang terdeteksi oleh robot. Pada gambar tersebut robot tetap melakukan penjajakan terhadap objek pertama. Gambar 5.17 (c) terjadi proses deteksi ulang yang menyebabkan perubahan ID pada tiap objek. Orang pertama (sebelah kanan) yang semula berwarna biru dan memiliki ID samadengan satu, berubah menjadi berwarna hijau dan ter-

jadi proses kalibrasi ulang. Sedangkan orang kedua (sebelah kiri) yang semula berwarna hijau dan mempunyai ID 2 berubah menjadi berwarna biru dan terjadi proses kalibrasi ulang.

Pada gambar 5.17 (d) objek di sebelah kanan yang sebelumnya berwarna hijau dan mempunyai ID 2, berubah menjadi berwarna kuning dan mempunyai ID 3. Untuk objek yang berada disebelah kiri tetap berwarna biru dan mempunyai ID 1. Dalam proses ini kedua objek dilakukan proses kalibrasi untuk menentukan objek tersebut manusia atau bukan. Gambar 5.17 (e) menunjukkan bahwa kedua objek berhasil mengalami proses kalibrasi.



Detik 11



Detik 13



Detik 15

**Gambar 5.18:** Hasil pengujian saat terdapat 2 orang



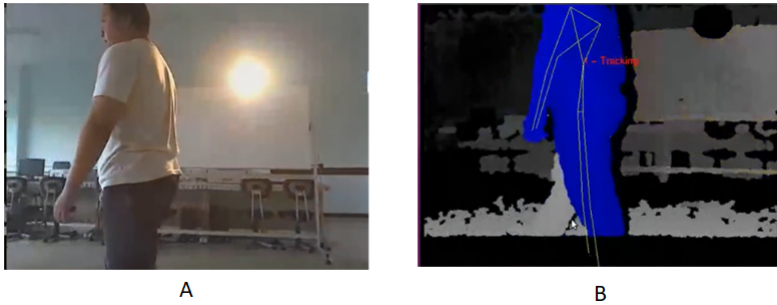
### 5.2.6 Pengujian Penjajakan ketika Terdapat Bayangan dari Objek

Pengujian ini bertujuan untuk mengetahui hasil penjajakan jika objek memiliki bayangan. Pada proses ini akan terlihat apakah bayangan dari objek tersebut juga ikut terdeteksi dan apakah sinar yang pencahayaan akan mempengaruhi proses penjajakan.



**Gambar 5.19:** Ilustrasi Pengujian

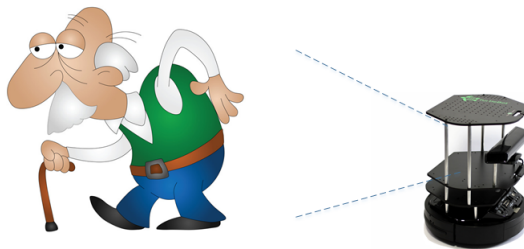
Lampu yang digunakan dalam pengujian ini merupakan lampu yang arah pencahayaannya dapat diatur (tidak menyebar) dan memiliki intensitas yang tinggi. Hasil pengujian menunjukkan bahwa adanya bayangan tidak mempengaruhi proses penjajakan dan sinar dari lampu tersebut tidak berpengaruh terhadap hasil tangkapan dari sensor. Sehingga robot dapat melakukan penjajakan dengan baik.



**Gambar 5.20:** Hasil yang didapat robot, A hasil citra RGB dan B hasil deteksi manusia dengan *depth*

### 5.2.7 Pengujian Penjajakan ketika Objek Berjalan Menggunakan Alat Bantu

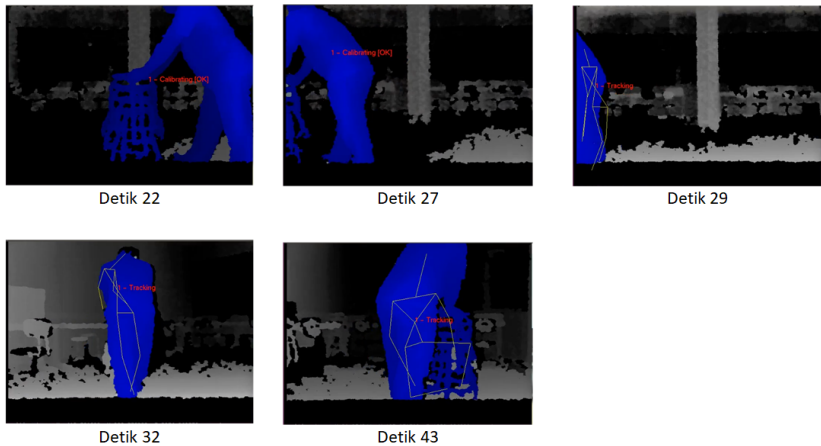
Pengujian ini dilakukan untuk mengetahui proses penjajakan ketika objek saat berjalan menggunakan alat bantu. Dalam pengujian ini apakah robot dapat mendeteksi dengan benar dan melakukan penjajakan dengan baik.



**Gambar 5.21:** Ilustrasi Pengujian

Hasil dari pengujian ini menunjukkan bahwa adanya alat bantu jalan membuat proses deteksi menjadi lebih lama. Gambar 5.22 menunjukkan proses deteksi dari sisi samping ketika objek menggunakan alat bantu. Pada gambar tersebut waktu yang digunakan untuk me-

netukan objek tersebut manusia atau bukan lebih lama dari pada ketika objek berjalan tanpa alat bantu. Namun ketika objek terlihat bagian belakang maka waktu proses deteksi sama seperti biasanya. Hal ini dikarenakan pada saat objek terlihat dari sisi belakang alat bantu yang digunakan objek tertutup oleh tubuh objek sehingga sistem melihat objek bergerak seperti tanpa alat bantu.

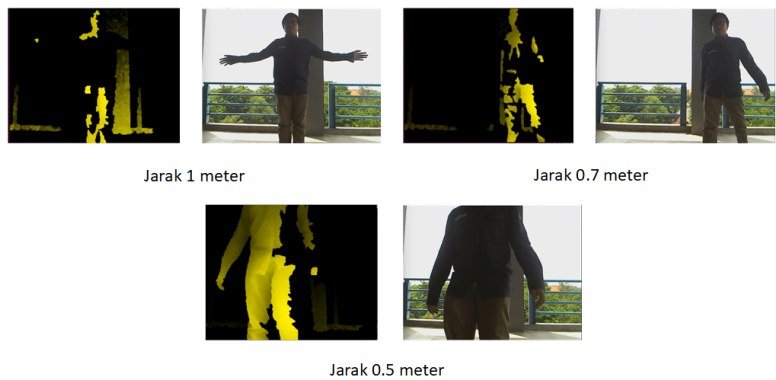


**Gambar 5.22:** proses deteksi ketika menggunakan alat bantu

### 5.2.8 Pengujian Penjajakan Diluar Ruangan (Outdoor)

Pengujian ini bertujuan untuk mengetahui jika proses penjajakan dilakukan di ruang terbuka. Pengujian ini dilakukan pada ruang terbuka yang terkena sinar matahari. Hasil pengujian ini menunjukkan bahwa ketika dilakukan pengujian diluar ruangan berpengaruh terhadap jangkauan sensor. Sensor yang ketika dalam ruangan dapat beroperasi dalam jarak 1 meter sampai 3 meter, ketika diluar ruangan jarak tangkapan sensor menjadi 0.5 meter hingga 1.5 meter (Gambar 5.23). Penyebab berkurangnya jangkauan sensor ini diakibatkan karena pantulan dari IR projector tidak dapat ditangkap oleh IR Sensor dengan sempurna akibat adanya gangguan atau pengaruh dari sinar matahari. Pada pancaran sinar matahari terdapat sinar *infrared* dan sinar *ultraviolet*. Sinar *infrared* merupakan

sinar yang digunakan pada sensor depth, sehingga dimungkinkan berkurangnya jangkauan deteksi akibat sinar dari sensor kalah dengan sinar dari matahari.



**Gambar 5.23:** Hasil deteksi diluar ruangan

# BAB 6

## PENUTUP

### 6.1 Kesimpulan

Tugas akhir ini telah berhasil mengimplementasi metode *object following* pada *robot service* dan berhasil melakukan penjajakan terhadap objek. Berdasarkan hasil pengujian yang didapatkan, kesimpulan yang dapat diambil adalah sebagai berikut :

1. Kecepatan maksimal yang dapat dilakukan oleh robot sebesar 1,6 m/s .
2. Berdasarkan pengujian berbagai pose disimpulkan bahwa metode *skeletal tracking* lebih baik dari pada metode HOG. Hal ini ditunjukkan pada hasil deteksi oleh HOG pada 9 pose memiliki keberhasilan sebesar 66.6%, sedangkan pada metode *skeletal tracking* keberhasilan sebesar 100%.
3. Berdasarkan pengujian kemampuan robot dalam melakukan penjajakan, robot dapat melakukan penjajakan terhadap objek yang sedang bergerak mengelilingi ruangan.
4. Berdasarkan respon kecepatan robot diperoleh, semakin besar jarak antara robot dan objek maka kecepatan gerak robot akan semakin cepat dan berlaku sebaliknya.
5. Hasil pengujian jarak diperoleh jarak ideal agar robot dapat mendeteksi objek manusia antara 1 meter sampai 2.5 meter.
6. Berdasarkan hasil pengujian *crossing people*, robot tidak dapat melakukan penjajakan hingga akhir.

### 6.2 Saran

Demi pengembangan lebih lanjut mengenai tugas akhir ini, disarankan beberapa langkah lanjutan sebagai berikut :

1. Pembutanan *map* atau *route path* agar robot dapat berjalan tanpa menabrak benda-benda disekitarnya.
2. Dalam pengembangan sistem ini dapat menambahkan beberapa perangkat sensor yang dapat digunakan untuk mendeteksi benda-benda yang berada disekitar robot.
3. Menerapkan metode lain sebagai metode pendeteksi manu-

sia, baik menggunakan sensor RGB atau menggunakan sensor Depth.

## DAFTAR PUSTAKA

- [1] gadgetflow, “Buddy-01.” <https://gadgetflowcdn.com/wp-content/uploads/2015/07/BUDDY-06.jpg>. (Dikutip pada halaman xi, 5).
- [2] pepper robot, “pepperhire1.” <http://pepperrobothire.com/wp-content/uploads/2016/08/pepperhire1.png>. (Dikutip pada halaman xi, 5).
- [3] yujin robot, “Kobuki spesifikasi.” [http://en.yujinrobot.com/wp-content/uploads/web\\_img/kobuki\\_spec.jpg](http://en.yujinrobot.com/wp-content/uploads/web_img/kobuki_spec.jpg). diakses 15 september 2017. (Dikutip pada halaman xi, 7).
- [4] robot advance, “Docking charger.” [https://www.robot-advance.com/EN/ori-turtlebot2-docking-station-1190\\_1531.png](https://www.robot-advance.com/EN/ori-turtlebot2-docking-station-1190_1531.png). diakses 29 september 2017. (Dikutip pada halaman xi, 8).
- [5] K. Team, “Kobuki user guide,” vol. 1, no. 1, 2017. (Dikutip pada halaman xi, 9, 10).
- [6] Microsoft, “Microsoft developer network.” <https://msdn.microsoft.com/en-us/library/hh973074.aspx>. diakses 14 November 2017. (Dikutip pada halaman xi, 12).
- [7] K. S. dan Supriyanto Sikumbang, “*Human Detection* menggunakan metode *Histogram of Oriented Gradients*,” Jurnal Pendidikan Teknik Elektro, vol. 4, no. 2, 2015. (Dikutip pada halaman xi, 17).
- [8] “Ukinect (openni+nite 2.x).” <http://www.doc.flash-robotics.com/index.php/software/software-modules/ukinectopenni2>. diakses tanggal 7 November 2017. (Dikutip pada halaman xi, 19).
- [9] Asus, “Asus xtion pro spesifikasi.” [https://www.asus.com/3D-Sensor/Xtion\\_PRO/specifications/](https://www.asus.com/3D-Sensor/Xtion_PRO/specifications/). diakses 14 November 2017. (Dikutip pada halaman xv, 13).

- [10] K. K. RI, “Gambaran kesehatan lanjut usia di indonesia,” vol. 1, no. ISSN 2088 - 270X, 2013. (Dikutip pada halaman 1).
- [11] S. P. Laksono, “ibm posbindu melati dan posbindu ceria. sistem informasi ageing monitoring medical record,” 2016. (Dikutip pada halaman 2).
- [12] I. F. of Robotic, “Service robot.” <https://ifr.org/service-robots/>. (Dikutip pada halaman 5).
- [13] yujin robot, “iclebo kobuki.” <http://kobuki.yujinrobot.com/about2/>. diakses 15 juni 2017. (Dikutip pada halaman 7).
- [14] howstuffworks, “How microsoft kinect work.” <https://electronics.howstuffworks.com/microsoft-kinect2.htm>. diakses tanggal 10 Januari 2018. (Dikutip pada halaman 11).
- [15] T. P. Fiqar, “Kontrol gerakan humanoid robot menggunakan sensor penangkap gerak,” 2013. (Dikutip pada halaman 11, 18).
- [16] A. C. Inc, “Asus xtion pro live.” [https://www.asus.com/us/3D-Sensor/Xtion\\_PRO\\_LIVE/](https://www.asus.com/us/3D-Sensor/Xtion_PRO_LIVE/). diakses tanggal 3 Agustus 2017. (Dikutip pada halaman 13).
- [17] ROS, “About ros.” <http://www.ros.org/about-ros/>. diakses tanggal 7 Agustus 2017. (Dikutip pada halaman 15).
- [18] N. D. dan Bill Trigs, “*histogram of oriented gradient for human detection*,” International Conference on Computer Vision Pattern Reconigition, 2005. (Dikutip pada halaman 17).
- [19] ROS, “Primesense releases drivers as open source, openni launched.” <http://www.ros.org/news/2010/12/primesense-releases-drivers-as-openni-launched.html>. diakses tanggal 10 November 2018. (Dikutip pada halaman 18).



- [20] Primesense, “Primesense nite algorithms 1.5,” (Dikutip pada halaman 20).

*Halaman ini sengaja dikosongkan*

## BIOGRAFI



Muhammad Ardi Pradana, Lahir pada 11 Agustus 1995 di Probolinggo, Jawa Timur. Penulis menempuh pendidikan SD hingga SMA di Probolinggo. Penulis menyelesaikan pendidikan SD di SD Taruna Dra. Zulaeha pada tahun 2007. Kemudian melanjutkan pendidikan SMP di SMP Taruna Dra. Zulaeha dan menyelesaikan pendidikan SMP pada tahun 2010. Selanjutnya ia menempuh pendidikan SMA di SMA Taruna Dra. Zulaeha dan menyelesaikan pada tahun 2013. Setelah lulus SMA, ia melanjutkan pendidikan di Institut Teknologi Sepuluh Nopember (ITS) Surabaya pada Departement Teknik Komputer bidang studi Telemtika.

Selama menempuh pendidikan di bangku perkuliahan, penulis aktif dalam UKM (Unit Kegiatan Mahasiswa) Robotika, baik sebagai anggota ataupun pengurus. Selain itu penulis juga tergabung dalam tim robot ITS selama 3 tahun. Sewaktu bergabung dengan tim robot ITS, penulis bergabung dengan tim Barunastra dan tim ichiro. Penulis mendapatkan penghargaan selama bergabung dengan tim robot ITS, penghargaan yang diperoleh dari tingkat regional, nasional, dan internasional. Selain itu penulis juga tercatat sebagai asisten Lab B201. Penulis sangat tertarik dengan hal yang berhubungan dengan elektronika, komputer, dan seni musik. Harapan penulis agar dapat mengembangkan diri dan dapat melakukan sesuatu yang bermanfaat bagi kehidupan.

*Halaman ini sengaja dikosongkan*